DOCUMENT RESUME

ED 184 874 SE 030 535

AUTHOR	Woolever, James
TITLE	Project SOUL: Computer Training Program for High School Students from Disadvantaged Areas. Part II, The Keypunch and Data Processing Courses. Technical Report.
INSTITUTION	University of Southern California, Los Angeles. Dept. of Electrical Engineering.
SPONS AGENCY	National Science Foundation, Washington, D.C.
REPORT NO	USCEE-402-B
PUB DATE	Apr 71
GRANT	NSF-GJ-0981
NOTE	132p.; For related documents, see SE 030 534 and ED 180 765. Appendix I marginally legible.
EDRS PRICE	MF01/PC06 Plus Postage.
DESCRIPTORS	Compensatory Education; *Computer Oriented Programs; Computers; *Computer Science Education; Data Processing; *Disadvantaged Youth; Educationally Disadvantaged; Educational Objectives; Experimental Programs; *Program Descriptions; Programing; *Secondary Education; Secondary School Mathematics; Secondary School Science; Summer Programs; Teaching Guides
IDENTIFIERS	Project SOUL

ABSTRACT

"Project SOUL" is a summer program in which disadvantaged high school students are given computer training. This section describes the content of the Keypunch and Data Processing Courses conducted in 1970. One or both of these descriptions include course objectives, course outline, teaching guide, and methods of instruction and evaluation. (MK)

****
* Reproductions supplied by EDRS are the best that can be made *
* from the original document. *
****

April 1971 USCEE 402B

U.S. DEPARTMENT OF HEALTH,
EDUCATION & WELFARE
NATIONAL INSTITUTE OF
EDUCATION

THIS DOCUMENT HAS BEEN REPRODUCED EXACTLY AS RECEIVED FROM THE PERSON OR ORGANIZATION ORIGINATING IT. POINTS OF VIEW OR OPINIONS STATED DO NOT NECESSARILY REPRESENT OFFICIAL NATIONAL INSTITUTE OF EDUCATION POSITION OR POLICY

"PERMISSION TO REPRODUCE THIS MATERIAL HAS BEEN GRANTED BY
Mary L. Charles
NSF

TO THE EDUCATIONAL RESOURCES INFORMATION CENTER (ERIC)."

ED184874

UNIVERSITY OF SOUTHERN CALIFORNIA

Technical Report

PROJECT SOUL:

Computer Training Program for High School Students from Disadvantaged Areas

PART II - The Keypunch and Data Processing Courses

James Woolever

Supported by the National Science Foundation
under Grant No. GJ-0981

ELECTRONIC SCIENCES LABORATORY

Engineering

SE 030 535

April 1971 USCEE 402B

Technical Report

PROJECT SOUL:

Computer Training Program for High School Students from Disadvantaged Areas

PART II - The Keypunch and Data Processing Courses

James Woolever

Department of Data Processing
Cerritos Community College

Supported by the National Science Foundation under Grant No. GJ-0981

PART II CONTENTS

THE KEYPUNCH AND DATA PROCESSING COURSES 1

- 1. Introduction 1
- 2. Section I - Keypunch Operation 2
  - a. Course Description and Objectives 2
  - b. Course Outline 3
  - c. Teaching Guide 4
- 3. Section II - Fundamentals of Business Data Processing and COBOL Programming 7
  - a. Course Description 7
  - b. Course Objectives 7
  - c. Course Outline 8
  - d. Methods of Instruction and Evaluation 9
  - e. Teaching Guide 10

APPENDICES

A. Organizational Chart
B. Schedule of Classes
C. Aptitude Test
D. Instructional Supplements - Keypunch Course
E. Laboratory Exercises - Keypunch Course
F. Preliminary Exercises - Business Data Processing Course
G. Examinations - Business Data Processing Course
H. COBOL Programming Assignments - Data Processing Course
I. Samples of Student Work - Data Processing Course

THE KEYPUNCH AND DATA PROCESSING COURSES

1. Introduction

The information contained herein is a report on the Data Processing Program of the 1970 Project Soul, as well as a guide for the development of curricula for similar programs on a national scale. It consists of two sections: 1. Keypunch Operations, 2. Fundamentals of Business Data Processing and COBOL Programming. It contains a detailed description of the courses taught, in addition to teaching methods and techniques used.

A total of fifty high school juniors and seniors with non-scientific backgrounds participated in the Data Processing Program. In general terms, the primary goal was to provide these young persons sound basic training in commercial data processing. The training, practical in its form, was designed to expand the horizons of the students in the educational and occupational frames of reference, and thereby, to motivate their interest in continuing education. Secondary goals were to prepare these young persons for employment in the areas of keypunch operations and data processing, and to establish and maintain a file of competent data processing personnel for recruitment by firms in the Los Angeles area which have indicated an interest in our trained people.

The keypunch course accommodated ten students and lasted four weeks. Two hours per day were devoted to lecture and two to laboratory work.

Forty students in groups of twenty took part in the six-week business data processing course. The first two weeks were spent on the fundamentals of business data processing, and the remaining four were devoted to an intensive course in COBOL programming utilizing the Honeywell H-200 Computer System. Again, two hours were spent in lecture and two in laboratory; at least one of the latter involved hands-on experience on the hardware.

It is a pleasure to acknowledge the following persons for their valuable contributions in the development of this segment of Project Soul: Norman Gale, Jules King, John Kelder, Richard Avery.

2. Section I - Keypunch Operation

a. Course Description and Objectives

This course has been designed to acquaint the student with the various processes of Key Punching in a business environment.

Prerequisite: That student be able to type at a speed of 40 words per minute.

Upon successful completion of this four-week course, the student will be able to:

1. State from memory at least six functions which a punched hole in a card can perform.
2. Describe correctly a unit record card in terms of the number of columns, and the names of the various punch positions.
3. Define a card field and calculate the number of columns required to punch a given field.
4. When questioned, locate and explain the use of:
    - Main Line Switch
    - Punching Station
    - Card Stacker
    - Keyboard
    - Program Control Unit
    - Card Hopper
    - Reading Station
    - Back Space Key
    - Chip Box
5. Set up a program control card, using the automatic functions available with this machine.
6. Correct and manually duplicate error cards.

7. Define the following program codes:

   Automatic Duplication
   Automatic Skip
   Alpha Shift
   Field Definition

8. State the purpose of the following:

   Program Drum
   Column Indicator
   Program Card

9. Remove and replace a program card.

10. Correctly operate the card punch under program control.

Text: International Business Machines Reference Manual 024-026 Keypunch.

b. Course Outline

I. Class Opening
II. Principles of Unit Record Accounting
III. The Unit Record Card
IV. Features of the Punched Card
V. Manual Punching Exercises
VI. Program Control Unit
VII. Additional Features of the Card Punch
VIII. Verifying Procedures
IX. Examination

c. Teaching Guide

LESSON ANALYSIS SHEET

LESSON TITLE: The Recording of Information.

LESSON OBJECTIVES: Discuss principle of the recording function through the use of the keypunch machine

REFERENCE: 3M-DATA PROCESSING--Volumes 1, 2, and 3.

MATERIALS: Chalk
Blackboard
Overhead foils

MOTIVATIONS: Student will be required to use keypunch machine in future data processing classes and possibly in employment.

PRESENTATION ITEMS	APPLICATION AND FEEDBACK

Segment Objectives:
Technical operations and control of the keypunch machine.

I. Processing Cycle.

1. Cycle carried out by using the various types of machines.

2. 5 principal processing steps used in a punched card installation.
    - A. Recording
    - B. Classifying
    - C. Calculating
    - D. Summarizing
    - E. Reporting

II. Recording

1. The recording function is performed by utilizing a machine called a keypunch.

2. Keypunch

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
A. Most widely used method of recording data.	1. What are the similarities and differences between the typewriter and a keypunch?
B. Point out similarities to keypunch and typewriter.	
3. Components of the keypunch.	
A. Keyboard (Pg. 48 awad)	
B. Card hopper 500 cards	2. What is the function of the card hopper?
C. Punching station 12 Vertical Position Punch Dies	3. What is the difference between a reading station and a punching station?
D. Reading station 12 brushes	
E. Card stacker 500 cards	
F. Column indicator Next column to be punched.	4. What does the "column indicator" indicate?
G. Program Control establishes automatic control of basic operation on the Keypunch. a. Automatic skip b. Auto Duplication c. Shifting of punching mode	
H. Four basic punches for prog control. a. 12 Field def. b. 11 Auto skip	

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
c. 0 Auto Dup. d. 1 Alpha shift e. used separately or in combination	
I. Program card a. Regular card b. Star wheels	5. What is a program card?
III. Accuracy in recording.	
1. Keypunch slow and costly	
2. Of utmost importance that cards are punched correctly	
IV. Verifier	
1. Looks like Keypunch	6. What is the difference between the keypunch and the verifier?
2. Operator re-keys from same source document.	7. Why is it important to ascertain the accuracy of data processing at the keypunch stage?
3. Machine uses sense pins	
4. Error indicated on the error column 3 tries. U Notch	
5. Card correct notch at upper right hand corner	

EVALUATION

Keypunch program
card assignment

3. Section II - Fundamentals of Business Data Processing and COBOL Programming

a. Course Description

This course will be divided into two parts. The first of which will cover the fundamentals of Business Data Processing. In Part One the student will learn the basic operation of the following unit record machines: IBM keypunch, IBM interpreter, and IBM card sorter. The second part will utilize the most frequently used procedural language for Business Application-COBOL. COBOL is a "Near-English" language which provides for thorough documentation of the program and enables programmers to be able to program on all major manufacturers' equipment with a minimum of adjustment. Business application will be analyzed, flowcharted, coded, and debugged using the computer.

Prerequisite: Satisfactory completion of programmer aptitude test.

b. Course Objectives

1. Introduce the students to the fundamentals of Data Processing and its application in business today.
2. Discuss principles of the recording function through the use of the keypunch machine.
3. Discuss the principles of sorting in both numeric and alphabetic sequence.
4. To introduce the student to the operation and control panel wiring of the card interpreter.
5. Discuss and use the current methods of computer problem flowcharting.
6. To develop the student's ability to work effectively with any modern third generation computer system.

7. To develop in the student confidence that he can work and communicate in any computer environment by applying the general principles of programming to the specific language that he may encounter.
8. To provide the challenge of applying and extending the student's ability in problem solving situations.
9. To further extend the student's knowledge of current Data Processing Techniques.

c. Course Outline

Part One

1. Class Opening
2. Fundamentals of Data Processing
3. The Unit Record Card
4. The Card Sorter
5. The Interpreter

Part Two

- A. Electronic Data Processing and COBOL
    1. Data Characteristics and Organization
    2. What is a Procedure?
    3. What is COBOL?
- B. The Procedure Division
    1. The Parts of a COBOL Source Program
    2. Basic Procedure Division Elements
    3. The Move, Add, Subtract, Multiply, and Divide Verbs
    4. The GO TO, Perform and Stop Verb
    5. The IF Statement
    6. Input and Output
- C. The Data Division
    1. The File Description
    2. Level Structure
    3. Record Description

    4. The Picture Clause
    5. Condition Name Value Entries
    6. The Working Storage Section
    7. The Constant Section
D. The Environment Division
E. The Identification Division

d. Methods of Instruction and Evaluation

Text and References:

1. Text: Spitzbarth, Laurel M., Basic COBOL Programming, Menlo Park, California, Addison-Wesley Publishing Co.
2. References: International Business Machine:

026-024	Keypunch
029	Keypunch
557	Interpreter
083	Card Sorter

Methods of Instruction:

1. Instructor Explanation and Demonstration of Course Content
2. Use of Overheads and Computer Listings to Provide Actual Models of Language and Techniques Discussed
3. Have Students Write Programs that Utilize All Aspects of Course
4. Class Discussion on Projec[illegible] Assignments

Methods of Evaluation:

1. Regular Attendance
2. Laboratory Projects
3. Formal Testing

e. Teaching Guide

LESSON ANALYSIS SHEET

LESSON TITLE: Introduction to the Fundamentals of Data Processing

LESSON OBJECTIVES: To introduce the students to the fundamentals of data processing and its applications in business today.

REFERENCE: 3M DATA PROCESSING -- Volumes 1, 2, and 3.

MATERIALS: Chalk
Blackboard
Overhead Foils

MOTIVATIONS: Indicate to the students that this is a new and growing field with many opportunities.

PRESENTATION ITEMS — APPLICATION AND FEEDBACK

Segment Objectives:
Discuss Data Processing course taught at Cerritos College.

I. Why should data processing be a field of study?

1. Complexity of business organization:
   - A. Early barter to corporate form.
   - B. Tribe leadership
     Gigantic government
2. Physical Factors:
   - A. Growth of customer force
   - B. Credit as a way of life
3. Cost Factors:
   - A. Cost of clerical help
   - B. Cost of inefficiency
   - C. Cost of not planning
4. Labor factor:

- A. Increase in 'white collar"
- B. Inefficiency in 'added personnel"
- C. Job enlargement.

5. Speed factor:
    - A. Currency of information
    - B. Best information is

II. What are the fields of Data Processing?

1. Manual:
    - A. Office Machines
    - B. Keysort systems
    - C. Microfilm

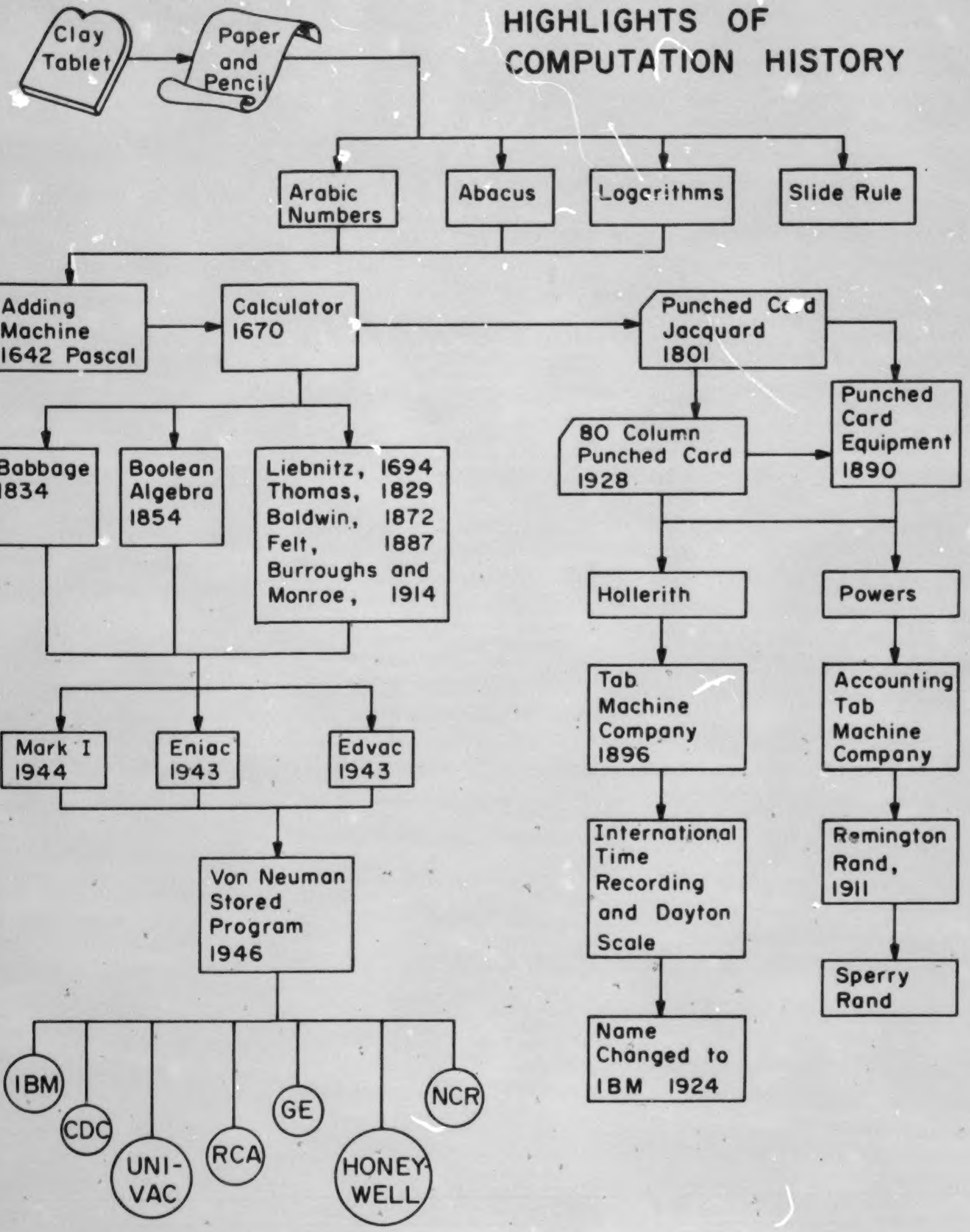
2. Mechanical (Punched Card)
    - A. Standard size record
    - B. Automatic manipulation

3. Electronic Data Processing
    - A. Computer System
    - B. Applications

EVALUATION

None

HIGHLIGHTS OF COMPUTATION HISTORY


Clay Tablet
Paper and Pencil
Arabic Numbers
Abacus
Logarithms
Slide Rule
Adding Machine 1642 Pascal
Calculator 1670
Punched Card Jacquard 1801
Punched Card Equipment 1890
80 Column Punched Card 1928
Babbage 1834
Boolean Algebra 1854
Liebnitz, 1694
Thomas, 1829
Baldwin, 1872
Felt, 1887
Burroughs and Monroe, 1914
Hollerith
Powers
Tab Machine Company 1896
Accounting Tab Machine Company
Mark I 1944
Eniac 1943
Edvac 1943
International Time Recording and Dayton Scale
Remington Rand, 1911
Von Neuman Stored Program 1946
Sperry Rand
Name Changed to IBM 1924
IBM
CDC
UNI-VAC
RCA
GE
HONEY-WELL
NCR

LESSON ANALYSIS SHEET

LESSON TITLE: Hollerith Coding System -- Unit Record Card.

LESSON OBJECTIVES: Understand coding system devised by Dr. Herman Hollerith and its application in modern business data processing.

REFERENCE: 3 M Data Processing -- Volume 1, 2, and 3.

MATERIALS:
Chalk
Blackboard
Overhead foils

MOTIVATIONS: Necessity of understanding the Hollerith coding system.

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
Segment Objective: Coding system, method of recording and unit record concept.	
I. Unit Record Card	
1. Holds one unit of information 2. Must be standard size (machine)	1. Why is it necessary to have a standard punched card?
II. Method of recording data on punched card.	
1. Data is transcribed from a source document to the unit record with the keypunch.	2. What is the role of the punched card in the recording stage?
2. Unit record becomes permanent storage.	3. What is the Hollerith Code?
3. Unit card can be duplicated when desired or necessary.	
III. Source of cards.	

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
2. Major company IBM (80) Remington Rand (90)	
IV. 80 Column Card Hollerith Code	
1. 80 vertical spaces called columns	
2. Columns are numbered from left to right 1--80	
3. Each column can only contain "1" character	
V. Punching Positions	
1. The 80 column unit record card is divided into two punching positions. A. ZONE B. DIGIT	
VI. Zone Position	
1. 3 horizontal rows at the top of the card.	4. What are the two punching positions of the 80 column card?
2. Two of which are used for zone punching only.	
3. One is a combination punch.	
VII. Digit Position	
1. 10 horizontal rows	
2. Represent position 0--9	
VIII. Edge and Face of card.	

PRESENTATION ITEMS

Sample handout.

1. Nine edge
2. 12 edge
3. Face

IX. Numeric recording of information.

1. One hole per column

   40875 require 5 columns

X. Alphabetic recording

1. Two holes per column
2. ZONE AND DIGIT SYSTEM

   A - I
   J - R
   S - Z

XI. Card Layout

1. Cards must have predetermined format.
2. Field A column or columns set aside for specific information.
3. Field max. 80, min. 1
4. Numeric Field must fill with zeros.
5. Alpha field
6. Fixed info left
7. Variable info right

XII. Card Design

APPLICATION AND FEEDBACK

5. What is meant by "face down 9-edge first"?

6. What is a field?

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
1. Uniform fields	
2. Identifying mark color stripes	7. How many different ways can a card be identified?
XIII. Control fields	8. What is a control field used for?
X-row used for control	

EVALUATION

Review Question

LESSON ANALYSIS SHEET

LESSON TITLE: Classifying information

LESSON OBJECTIVE: To discuss the principle of sorting of data in both numeric or alphabetic sequence using the 82 Sorter.

REFERENCE: 3M-DATA PROCESSING, Volumes1,2 and 3.

MATERIAL: Chalk
Blackboard
Overhead foils

MOTIVATION: Emphasize the importance of the classifying process within data processing installations.

PRESENTATION ITEMS	APPLICATION AND FEEDBACK

Segment Objectives:

To introduce student to the funcmentals of classification of data using sorting machines.

Classifying Information

I. Classifying
   1. Process in which like transactions are grouped, or arranged together, in either alphabetic or numeric.

II. Three basic types of classification
   1. Sorting in sequence
   2. Selecting
   3. Grouping

III. Sequence
   1. Both numeric and alphabetic

IV. Selecting

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
Operation which involves pulling certain cards from a file without disturbing the sequence of the remainder of the file.	1. What is an example of selecting?
V. Grouping Sorting by common characteristics.	
1. By branch 2. Regional office	2. What is an example of Grouping?
VI. The Sorter 1. Fastest machine in punched card data processing installation. 2. 13 pocket 1. will accept any possible punch & blank. 3. Only one brush that is positioned over column being read.	
VII. Numeric Sorting 1. Low order position to high order position	
VIII. Alphabetic sorting 1. Low order to high order position	3. Define "Reverse digit method" of sorting
2. Two pass on each column 3. Numeric 1st and Zone 2nd	
IX. Compilation of sort time	4. Why is it important to compute sort time?

$$\frac{\text{Volume + Numeric Columns}}{\text{Speed or Sorter}}$$

+ 25% handling

LESSON ANALYSIS SHEET

LESSON TITLE: Introduction to the IBM 548 Interpreter

LESSON OBJECTIVES: To introduce the student to the operation and control panel wiring of the 548 interpreter

REFERENCES International Business Machines Reference Manual 548 Interpreter

MATERIALS: Projector Foils
- 548 Machine
- Card Path
- Control Section
- Type Bar Design
- Control Panel Functions
- Wiring Diagram Sheet

MOTIVATION The student will be introduced to the wiring concept using the 548 in an attempt to eliminate any confusion as to what the control panel accomplishes

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
Segment Objectives To introduce the machine functions of the 548	
1. Interpreter will print information on same card a) increases value of card b) easy to read document	1. Why do we interpret cards? 2. Why would it be helpful in maintaining a file?
2. Useful for file maintenance a) proofreading b) prepunched tub file	3. How long would it take to interpret 8 000 cards?
3. Speed of interpreter is 60 cards per minute (Foil#1)	

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
Segment Objectives: To introduce the operating features of the 548	
1. Main line switch (Foil #2)	1. Is warm-up required?
2. Start and stop keys.	2. How long must the start key be depressed?
3. Hopper a) cards feed face up, 12 edge first. b) feed from back file.	3. How many cards will the hopper hold?
4. Stacker.	4. What are the problems in feeding cards face up?
5. Path of card through machine. (Foil #3)	5. How many read stations on the 548?
Segment Objectives: Discuss printing characteristics.	
1. Two lines of printing possible.	1. Where are the two lines printed?
2. Printing position knob. (Foil #2).	2. Where is the print knob located?
3. Sixty type bars. (Foil #4).	3. How do we interpret 80 columns with only 60 type bars?
4. Control panel functions. (Foil #%)	4. Why do we have a control panel?
5. X brushes. a) reason for them. b) how to set them.	5. How many X brushes a standard?
Segment Objectives: To introduce the control panel and the principles of wiring.	
1. Control panel wiring.	1. How many reading brushes are

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
a) reading brushes exit hubs	there on the 548?
b) typebars entry hubs.	
c) common hubs.	2. Explain the term Common Hubs.
2. Wiring to print in two places.	3. What is offset wiring?
3. X eliminators. (Foil # 6)	4. How many X eliminators are standard on the 548?
	5. How does the interpreter differ from most IBM machines?

LESSON ANALYSIS

LESSON TITLE: Block Diagramming

LESSON OBJECTIVE: Discuss and use current methods of Computer Block Diagramming.

REFERENCE: 3M-Data Processing-Volumes1,2,3, and illustrations

MATERIALS:
Blackboard
Chalk
Overhead foils
Handout materials.

MOTIVATION: A program is a part of a problem solving cycle which, when done carefully, saves time.

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
Segment Objective: To introduce student to scientific method of problem solving.	
I. Electronic Computer is dependent on man.	
1. Will aid man in solving problem.	1. What is involved in data organization?
2. It is only as useful as the Programs that direct its work.	
II. Preparation and planning.	
1. Overall understanding of problem.	2. Why is the problem-definition stage important in problem solving?
2. Proper planning before implemented.	
III. Flowchart. Definition: A Flowchart is a chart depicting the flow of data and operations in a D.P. system.	

PRESENTATION ITEMS	APPLICATION AND FEEDBACK
IV. Elements of the flowchart. 1. Problem solving process.	3. What is a block diagram?
2. 5 parts. 1. Define the problem. 2. Organize the require data. 3. Devising a procedure for a desired solution. 4. Testing procedure 5. Carrying out the program	4. What is a flowchart?
V. Define Problem. 1. Determine what input-output a. Mention form layout. b. What info in data cards. c. Source documents.	5. What is branching? When and why is it used in a block diagram?
2. Must be sufficient amount of time allowed. 1. Well organized program 2. Short and sloppy program 3. Awareness of the limitations of both the equipment and personnel. 1. Computer size, etc. 2. Advance-Trainee Programer	6. What is involved in the debugging stage?
VI. Data Organization. 1. Arrangement of component parts. 2. Logical arrangement of data from various areas.	7. What is the difference between a logic error and a clerical error?
VII. Development Stage (Devise Procedure) 1. Devising procedure - based and data organization. 2. All necessary steps involved in reporting.	
VIII. Block diagram.	

1. Visual Aid.
2. # of Block containing instruction.
3. A block diagram is essential in that it makes it possible to write the program in an orderly manner.
4. Example of Block diagram:

STAND
UP

SPEAK
UP

SHUT
UP

5. Detailed Flow-chart.
   Reference page 220.
   Discuss how to get up in the morning.

IX. Symbols.
1. Symbols vary according to manufacture.
2. Represent different ideas.
3. Direction of flow

4. Input-Output Symbol

Read Card Input	Punch Card Output

5. Processing Symbol.

Add, Subtract, Move, etc.

6. Reference figure 18.5 page 224 and discuss LOOP.
7. Decision Symbol.

   Alternative operation
      EQUAL - UNEQUAL
      HIGH, LOW

8. Branching
   Detour around certain instructions reference fig. 18.7, page 227

X. General Diagraming Hints.
1. Step begin at the top and come down and toward the right.
2. Consistent use of symbols.
3. Minimize and clarify writing.

XI. Coding stage
1. After diagram is debugged
2. Translation of flowchart into the language of the Computer
   1. Assembly, computer, machine language.

XII, Debugging
1. To locate and correct errors in program.
2. To main type of errors
   1. Logic errors - Decision:

2. Cleric errors - Misspell
   Print PRRNT

XII. (Testing stage & Carrying Out Program)

1. Run test data.

LESSON ANALYSIS SHEET

LESSON TITLE: Data Organization

LESSON OBJECTIVE: To Discuss the Organization of Data

REFERENCE: A Guide to COBOL Programming: Daniel D. McCracken

MOTIVATION: Emphasize the Importance of Data Organization

PRESENTATION	APPLICATION AND FEEDBACK

DATA CHARACTERISTICS

1. Data refers to any information that can exist within a computer system.

2. Types of Data — What is a master-file?

   a. source
   b. master-file
   c. sorted
   d. edited — What does editing mean?

3. The most elementary unit of data is a character. Which may be a numeric or alphabetic character.

4. Data Items is composed of one or more characters.

5. Data Name is a symbol by which a data item is referred to. — What is the difference between a data name and a data item?

6. A record is composed of data items. These items are related.

   Example:

   Inventory File (record)

$$\left.\begin{array}{l}\text{Part number}\\ \text{Quantity on Hand}\\ \text{Quantity or Order}\end{array}\right\}\ \text{Data Items}$$

7. A File consists of records.

   a. Input files
   b. Output files

Evaluation:

Oral Discussion.

LESSON ANALYSIS SHEET

LESSON TITLE: What is a Procedure

LESSON OBJECTIVE: To Define a Procedure

REFERENCE: A Guide to COBOL Programming: Daniel D. McCracken.

MOTIVATION: To show the student the importance of Procedures.

PRESENTATION — APPLICATION AND FEEDBACK

SEGEMENT OBJECTIVE

To learn how to integrate the needs of the business. The characteristics of the data, and the powers and limitation of computers.

1. Procedures must be general so they will work correctly with data of like type.
2. Computer cannot exercise judgement unless it has been provided with explicit directions for making a decision.
3. The computer must be told how to know when it is finished.
4. Describe the tasks and consider the general approach to the computer procedure for carrying out the desired processing.

EVALUATION

Oral Discussion.

LESSON ANALYSIS SHEET

LESSON TITLE: What is COBOL?

LESSON OBJECTIVE: To Discuss the COBOL Compiler Language.

REFERENCE: A Guide to COBOL Programming: Daniel D. McCracken

MOTIVATION: The student will understand the elements of the COBOL language.

PRESENTATION	APPLICATION AND FEEDBACK
Segment Objective To discuss the COBOL Language which grew out of the requirement for a language that would be compatible with various computer systems.	
1. A computer cannot follow a procedure just as it stands; there must be an intermediate translation step first	What is a translation step?
2. To accomplish this translation the COBOL language is divided into a number of divisions. A. Procedure Division B. Environment Division C. Identification Division D. Data Division	
3. These divisions combined represent a COBOL program the translation of this program is called a compilation.	

4. The output from a COBOL compilation is known as the object program and is in a usable form for the computer.

5. The COBOL program after compilation is then ready to run data for report preparation.

Evaluation

Oral Discussion.

LESSON ANALYSIS SHEET

LESSON TITLE: Parts of a COBOL source program

LESSON OBJECTIVE: To discuss and use the various elements of a COBOL program.

REFERENCE: A Guide to COBOL Programming: Daniel D. McCracken

MOTIVIATION: To assist student understanding of the COBOL language.

PRESENTATION APPLICATION AND FEEDBACK

1. Identification Division is a new line to identify the program, the author, and data written.

```
Identification Division.
Program-ID. Demo.
Author. H. White.
```

2. Environment Division specifies the computer to be used, both for the compilation and for the object program; the two are ordinarily the same.

```
Environment Division.
Configuration Section.
Source-Computer. H-200-Special
Object-Computer. H-200, Supervisor,
                 No Segmentation.
```

3. Data Division includes the files, records, and data items that are to be processed or produced as results.

```
Data Division
Work-Storage Section.
01 Name-Record-File.
    02 Name-In Picture x(20).
    02 Address-In Picture x(20).
    02 City-In Picture x(20).
    02 Zip-In Picture 9(s).
```

4. Procedure Division specifies the steps the programmer wishes the computer to carry out in processing the data.

```
Procedure Division.
   Display CΔC.

Go-File.
        Accept Name-Record-File.
        Display Name-Record-File.
```

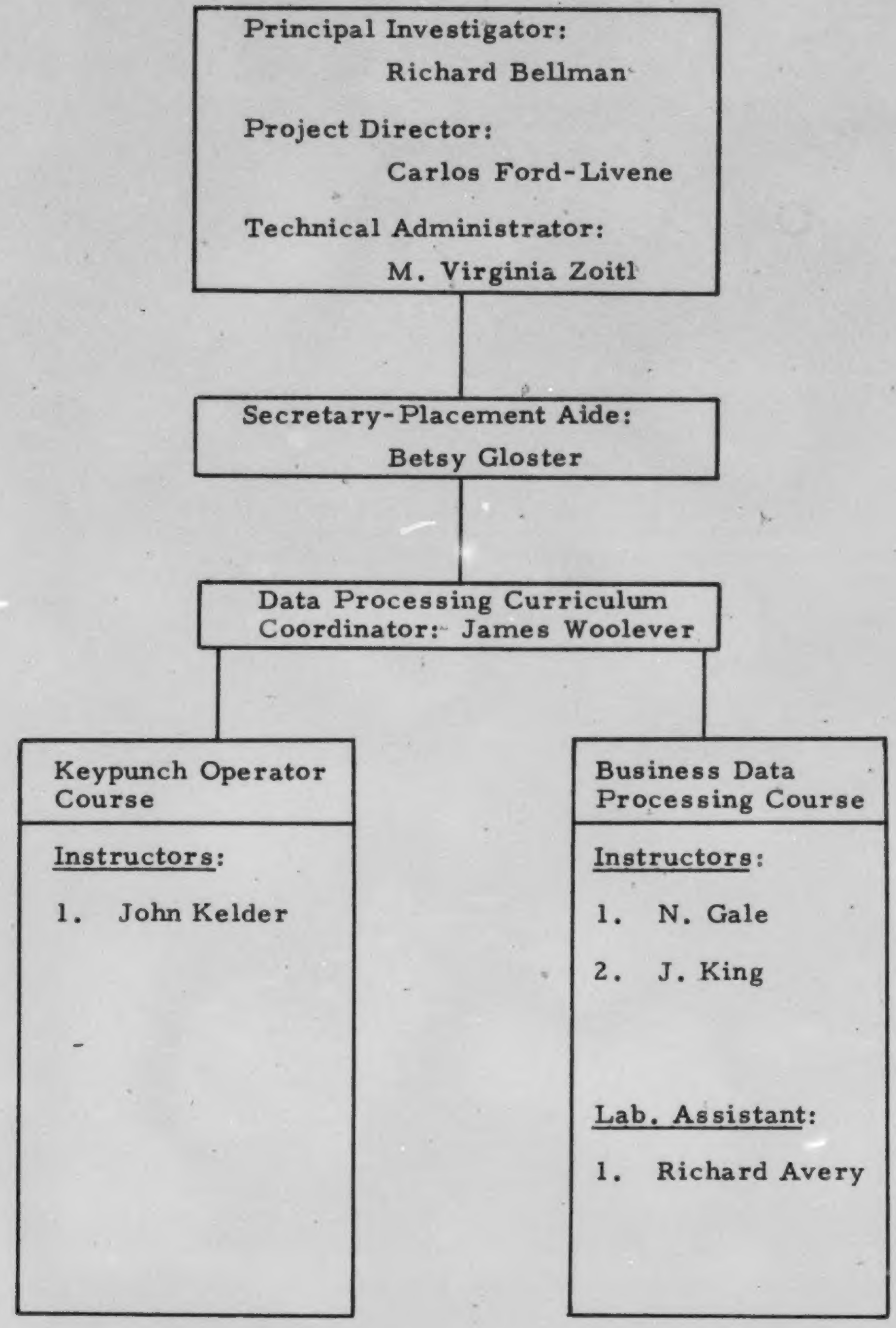
Evaluation

Oral Discussion

APPENDIX A

Organizational Chart

"PROJECT SOUL" 1970


Principal Investigator:
Richard Bellman
Project Director:
Carlos Ford-Livene
Technical Administrator:
M. Virginia Zoitl
Secretary-Placement Aide:
Betsy Gloster
Data Processing Curriculum Coordinator: James Woolever
Keypunch Operator Course
Instructors:
1. John Kelder
Business Data Processing Course
Instructors:
1. N. Gale
2. J. King
Lab. Assistant:
1. Richard Avery

APPENDIX B

Schedule of Classes

COURSE	DATE	CLASS SIZE	TIME	LOCATION	INSTRUCTOR	KEYPUNCH DISTRIBUTION
Keypunch	June 29-July 24	10	10-12 1-3	OHE 134	John Kelder	OHE 132-5kps
Business Data Processing, Section A	June 29-August 7	20	10-12 Lab. 1-3 Lect.	CSL 113 CSL 129	Norman Gale R. Avery (Lab. Assistant)	CSL-5kps
Business Data Processing, Section B	June 29-August 7	20	10-12 Lect. 1-3 Lab.	CSL 129 CSL 113	Jules King R. Avery (Lab. Assistant)	CSL-5kps

OHE denotes Olin Hall of Engineering

CSL denotes Computer Science Laboratory

APPENDIX C

Aptitude Test

INSTRUCTIONS FOR PART I

In Part I you will be given some problems like those on this page. The letters in each series follow a certain rule. For each series of letters you are to find the correct rule and complete the series. One of the letters at the right side of the page is the correct answer. Look at the example below.

		(1)	(2)	(3)	(4)	(5)
W.	a b a b a b a b	a	b	c	d	e

For this problem, the series goes: ab ab ab ab
The next letter in the series is a. Choice 1 is the correct answer.

		(1)	(2)	(3)	(4)	(5)
X.	a a b b c c d d	a	b	c	d	e

In Example X above, the series goes like this: aa bb cc dd. The next letter in the series is e. Choice 5 is the correct answer.

Now do Example Y below.

		(1)	(2)	(3)	(4)	(5)
Y.	c a d a e a f a	d	e	f	g	h

In Example Y, the series goes: ca da ea fa. Therefore, the correct answer is g, Choice 4.

Now do Example Z.

		(1)	(2)	(3)	(4)	(5)
Z.	a x b y a x b y a x b	a	b	c	x	y

In Example Z, the series goes like this: axby axby axb. Therefore, the correct answer is y, Choice 5.

In the problems on the following page, you are to select the correct letter on the right-hand side of the page which belongs next in the series. Indicate the correct answer on the answer sheet.

PART I

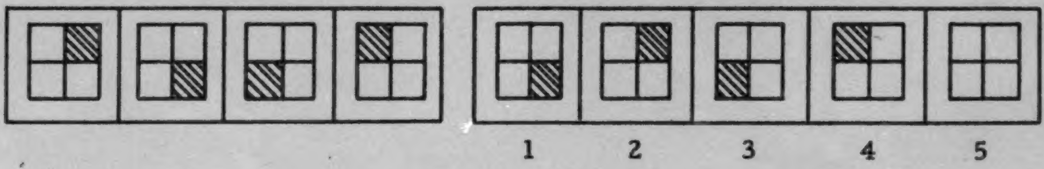
		(1)	(2)	(3)	(4)	(5)
1.	c d e x y z f g h x y z	i	j	k	l	m
2.	t s r t s r t s	r	s	t	v	w
3.	a b c c d e f f g	e	f	g	h	i
4.	m n m n k l o p o p k l	k	o	p	q	r
5.	a b c i j d e f i j	g	h	i	j	k
6.	a i b c i d e f	e	f	g	h	i
7.	a g b h c	d	f	g	h	i
8.	a e d h g	h	i	j	k	l
9.	k s j t i u h	v	w	x	y	z
10.	n j f m i e l	d	h	i	j	m

INSTRUCTIONS FOR PART II

In Part II you will be given some problems like those on this page. Each row is a problem. Each row consists of four figures on the left-hand side of the page and five figures on the right-hand side of the page. The four figures on the left make a series. You are to find out which one of the figures on the right-hand side would be the next or the fifth one in the series. Now look at Example X.

Example

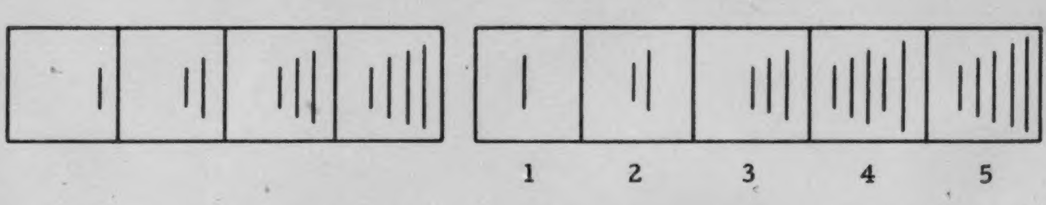
X



In Example X there is a clockwise movement of the striped square: upper right, lower right, lower left, upper left. The next or fifth position in this clockwise movement would thus be upper right, and so Choice 2 is the correct answer.

Now look at Example Y.

Y

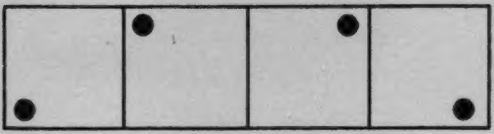


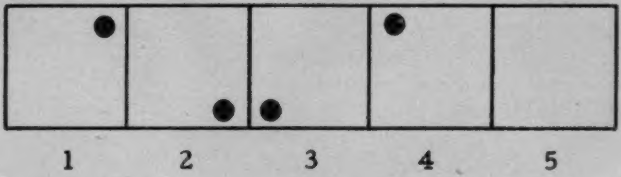
In the series of figures on the left, there is one more line in each box and these lines increase in length. Now look at the five choices on the right-hand side of the page and determine the correct answer.

You should have selected Choice 5 which has five lines, one more than the last box on the left with the fifth line slightly longer than the last line in Box 4.

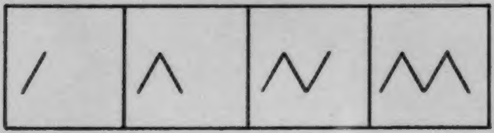
PART II

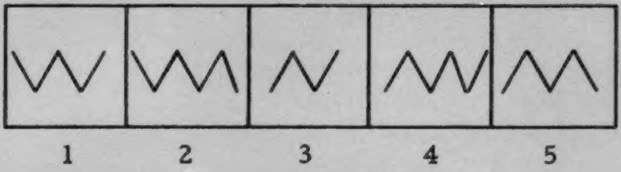
11.



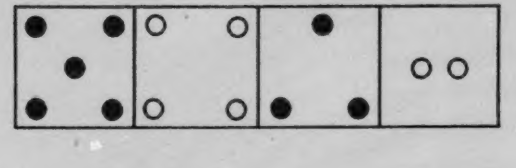


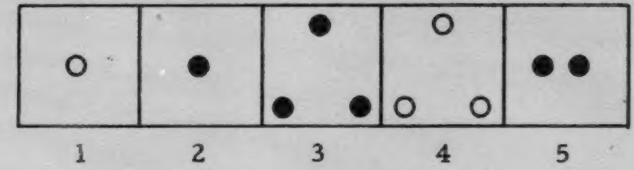
12.



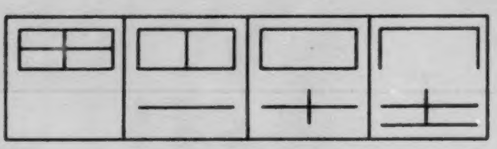


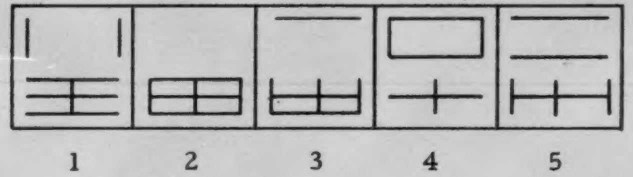
13.



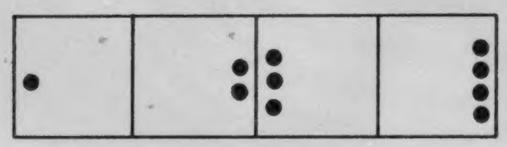


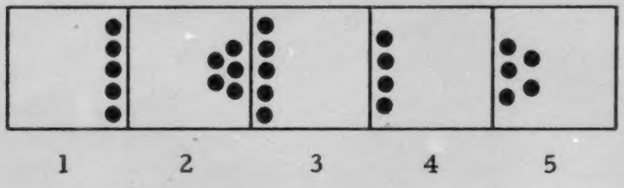
14.



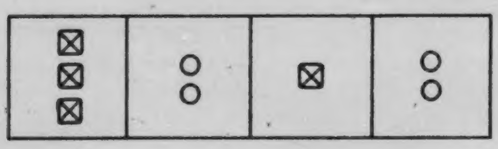


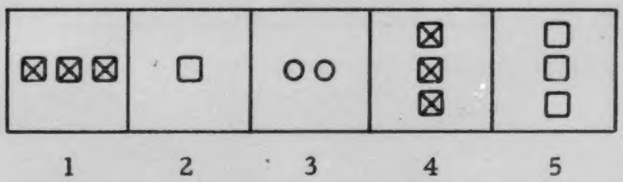
15.



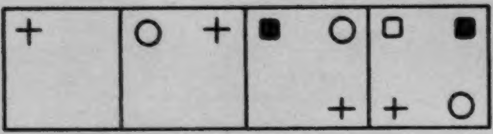


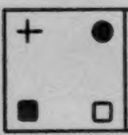
16.

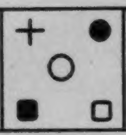


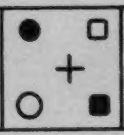


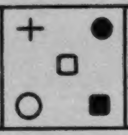
17.

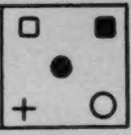



1

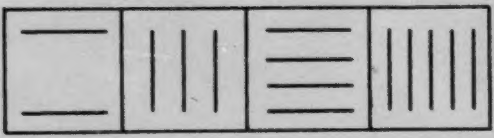

2

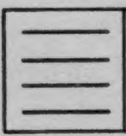

3

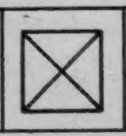

4

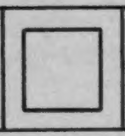

5

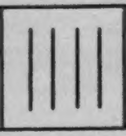
18.

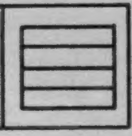



1

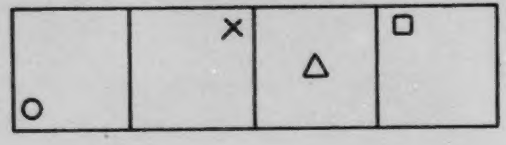

2

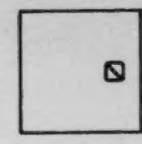

3

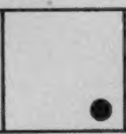

4

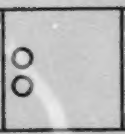

5

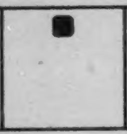
19.

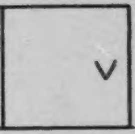



1

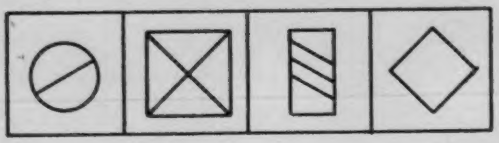

2

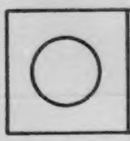

3

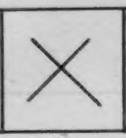

4

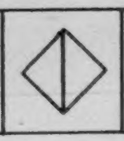

5

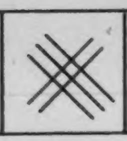
20.

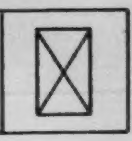



1


2


3


4


5

INSTRUCTIONS FOR PART III

In Part III you will be given some problems in arithmetical reasoning. After each problem there are five answers, but only one of them is the correct answer. You are to solve each problem and indicate the correct answer on the answer sheet. The following problems have been done correctly. Study them carefully.

Example X: How many apples can you buy for 80 cents at the rate of 3 for 10 cents?

(1) 6 (2) 12 (3) 18 (4) 24 (5) 30

The correct answer to the problem is 24, which is Choice (4).

Example Y: In 4 weeks John has saved $2.80. What have his average weekly savings been?

(1) 35¢ (2) 40¢ (3) 50¢ (4) 70¢ (5) 80¢

The correct answer to the above problem is 70¢, Choice (4).

PART III

21. A clerk multiplied a number by ten when it should have been divided by ten. The answer he got was 100. What should the answer have been?

(a) 1	(b) 10	(c) 100
(d) 1000	(e) 10,000	

22. The average salary of three programmers is $95 per week. If one programmer earns $115, a second earns $65, how much is the salary of the third programmer?

(a) $95	(b) $105	(c) $115
(d) $160	(e) $180	

23. If a card punch operator can process 80 cards in half an hour, how many cards can she process in a seven and one-half hour day?

(a) 560	(b) 600	(c) 800
(d) 1120	(e) 1200	

24. In a programming team of 12 persons, 1/3 are women and 2/3 are men. To obtain a team with 20% women, how many men should be hired?

(a) 4	(b) 6	(c) 8
(d) 12	(e) 20	

25. It cost a college 70 cents a copy to produce the program for the homecoming football game. If $15,000 was received for advertisements in the program, how many copies at 50 cents a copy must be sold to make a profit of $8000?

(a) 14,000	(b) 35,000	(c) 46,000
(d) 75,000	(e) 115,000	

APPENDIX D

Instructional Supplements - Keypunch Course

IBM KEYPUNCH - GENERAL INFORMATION

All IBM cards have 80 columns; they are read from left to right.

In each column there are 12 possible punching positions:

12
11 (or X) } zone punches + 1 through 9 digit punches = 12 positions.
0

The combination of 1 zone punch and 1 digit punch makes an alphabetic character.

12 zone punch + 1-9 digit punch = letters A-I
11 zone punch + 1-9 digit punch = letters J-R
0 zone punch + 2-9 digit punch = letters S-Z

The following chart is convenient for interpreting alphabetic information punched in an IBM card:

	1	2	3	4	5	6	7	8	9
12	A	B	C	D	E	F	G	H	I
11	J	K	L	M	N	O	P	Q	R
0	/	S	T	U	V	W	X	Y	Z

A field is a column or a group of columns, set aside by vertical lines, for one type of information which corresponds to the source document.

A dotted line in a field indicates the punctuation; such as the position of the decimal in a money or percentage field.

Corner cuts in cards enable the operator to see that all the cards are facing the correct way.

Only the left hand lower corner cut will not feed through the machine.

Fan and joggle the cards before placing them in the hopper.

Cards are placed in the hopper face forward, 9-edge down.

Cards which have no fields are called General Purpose cards, or 5081.

Cards which have fields are called Detail cards.

The card which is placed around the drum is called a program card.

When mounting a program card on the drum, first fasten the column 80 edge of the card under the smooth edge of the clamping strip.

When the Star Wheels are UP the machine is in alphabetic shift; when the Star Wheels are Down on a program card, the machine is in program control. (The program Unit controls, skipping, duplicating, alphabetic and numeric punching.)

Cards are placed in the hopper, face forward, 9-edge down.

The machine fuses are located behind the chip box.

The card column indicator points to the next column to be punched.

The card gauge should be used to check the punching registration, daily and after card jams.

Depression of the feed key does not cause feeding when a card has been registered for punching.

With automatic feed ON, when column 80 of a card passes the punching station, that card moves to the reading station, a card at the right in the card bed is positioned for punching, and another card is fed from the hopper.

The pressure roll release lever is depressed to permit the removal of a card caught at the punching or reading station.

A card can be manually inserted, at the right of either the punching or reading station.

When mounting a program card on a drum, first fasten the column 80 edge of the card under the smooth edge of the clamping strip.

When alphabetic information does not fill the entire field, the unpunched portion is normally skipped, by the use of the skip key, in conjunction with program card coding.

Cards in the punching and reading station, and the program card moves backward when the backspace key is depressed.

With program control ON, depression of the dash skip or dash key, always punches an X (11 punch), but causes skipping over numerical fields only.

Automatic skipping can be accomplished, when the program card is properly punched, and the automatic skip switch is ON.

Duplication by use of the duplicate key is faster with program control than without.

Depression of the duplicate key locks the machine if the column of the card at the reading station is unpunched and the keyboard is in numerical shift.

When numbers are to be occasionally punched in an otherwise alphabetic field, (such as street address), the program card is normally coded for alphabetic punching and the numerical shift key is depressed when necessary.

When date is automatically duplicated, to change the date, the automatic duplicate switch must be turned OFF, and the new information punched in the first detail card of the new date group.

The backspace key backspaces the cards continuously, as long as it is held depressed.

When a card is released due to a punching error, the information in a field programmed for automatic duplication is automatically duplicated regardless of its location.

When correcting an error during punching under program control, the operator may duplicate all of the correctly punched fields, and must rekey only the field containing the error column.

In order to punch alphabetic information with program control ON, either the program card must be coded 1 or the alphabetic shift key must be depressed.

IBM 024 CARD PUNCH

With Star Wheels UP this machine punches alphabetic information. In order to punch numeric information with the Star Wheels UP (program control off) the numeric key (NUM) must be held down.

THE PARTS OF THE MACHINE ARE AS FOLLOWS:

Main Line Switch (off and on)

Card Hopper

Pressure Plate (holds cards in hopper)

Card Beds (three) - punching station, reading station, and card stacker

Card Weight

Key Board (movable)

Jog Plates

Control Switches

Reading Board

Chip Box (empty daily or when necessary)

Fuses (two small ones may be replaced by operator)

Back Space Key

Program Control Lever (Star Wheels) - never put the Star Wheels down unless there is a card on the program drum

Pressure Roll Release Lever (press down the release cards caught in punching or reading station)

Card Column Indicator (sits on next column to be punched)

Keys - gray keys are punching keys, blue keys are functional keys, and keys with shadow in back of them are numeric keys

- Use first finger for 7, 4, 1, and DUP
- Use middle finger for 8, 5, 2 and X SKIP (the X SKIP or DASH SKIP key always punches an X (11 zone punch) and skips over numeric fields only)
- Use ring finger for 9, 6, 3 and 0.

Program Drum - 80-edge of card goes under smooth edge of clamping strip

IBM 024 KEY PUNCH

MAIN LINE SWITCH: This switch turns the machine ON and OFF.

CARD HOPPER AND PRESSURE PLATE: Inside the card hopper there is a pressure plate which holds the cards in position.

PUNCHING STATION: Punching is performed at the first right-hand station.

READING STATION: Cards are read for duplicating at the left-hand station.

CARD STACKER and CARD WEIGHT: Inside the card stacker is a card weight which holds the cards in position.

CARD BEDS: These are located to the right of the punching and reading stations and below the card stacker.

BACK SPACE KEY: This key is located below the card bed between the reading and punching stations.

KEYBOARD and READING BOARD: The keyboard is movable and sits on the reading board.

CONTROL SWITCHES: These switches are on top of the keyboard and they control automatic feeding of the cards and automatic skipping and duplicating. (The 026 Key Punch machine has a switch which controls printing also.)

JOG PLATES: These are located on both sides of the keyboard.

CHIP BOX and FUSES: The chip box is located under the reading board and is emptied daily. The machine fuses are located behind the chip box. (The two small ones may be replaced by the operator.)

PROGRAM CONTROL LEVER: This lever is located below the program unit and is used to raise and lower the Star Wheels. (Never lower the Star Wheels if there is no card on the program drum, nor pull the drum off when the Star Wheels are down.)

PROGRAM DRUM: The program drum is located inside the program unit.

COLUMN INDICATOR: This indicator, located at the base of the program drum holder, indicates to the operator the NEXT column to be punched.

PRESSURE-ROLL RELEASE LEVER: This silver lever, located above the column indicator and to the right of the program drum, is depressed to permit the removal of cards caught at the punching and reading stations.

KEYS: The gray keys are punching keys, the blue keys are functional keys, and the shadowed gray keys are the numeric keys. For touch system use:

Index finger for digits 7, 4, 1, and DUP
Middle finger for digits 8, 5, 2, and -SKIP-. (The DASH SKIP or X SKIP key always punches an X (11 zone punch) and skips over numeric fields only.)
Ring finger for 9, 6, 3, and 0.

HOW TO START KEY PUNCHING (FINGER EXERCISES)

1. Turn on the Main Line Switch.

2. Get a handful of general purpose cards (those having no fields) from the shelf.

3. Fan and joggle these cards then place them in the hopper of the machine face forward, 9-edge down.

4. Put the program card on the program drum, 80-edge under the smooth edge of the clamping strip.

5. Place the program drum on the spindle so that it rests firmly on the card column indicator, then put the Star Wheels down (program control lever on).

6. Press the Release Key (marked REL). If the program drum turns around, the machine is warm and you may continue to the next step. If the drum does not turn, wait until it does turn before continuing.

7. Press key marked FEED twice and then turn the Auto Feed and Auto Skip and Dup switches to the "on" position.

8. You may now punch the finger exercises. Be sure you are using the correct fingers as shown on your chart. The numbers are grouped in units of four for easier reading but there are no spaces between these groups; the numbers are punched consecutively.

9. If you make a mistake and want to start over, press the key marked REL (Release Key) and the error card will move to the read station, the card in the punch station will be registered for punching, and another card will be fed from the hopper. When this new card is completed, the error card will be in the stacker and must be removed. (Only correct cards are supposed to remain in the stacker.)

10. If you are leaving your machine for a period of time (coffee break), clear your machine by turning off the Auto Feed and Auto Skip and Dup switches and by pressing first the Release Key (REL) and then the Register Key (REG) alternately until the cards from the card beds are in the stacker. Then turn off the main line switch.

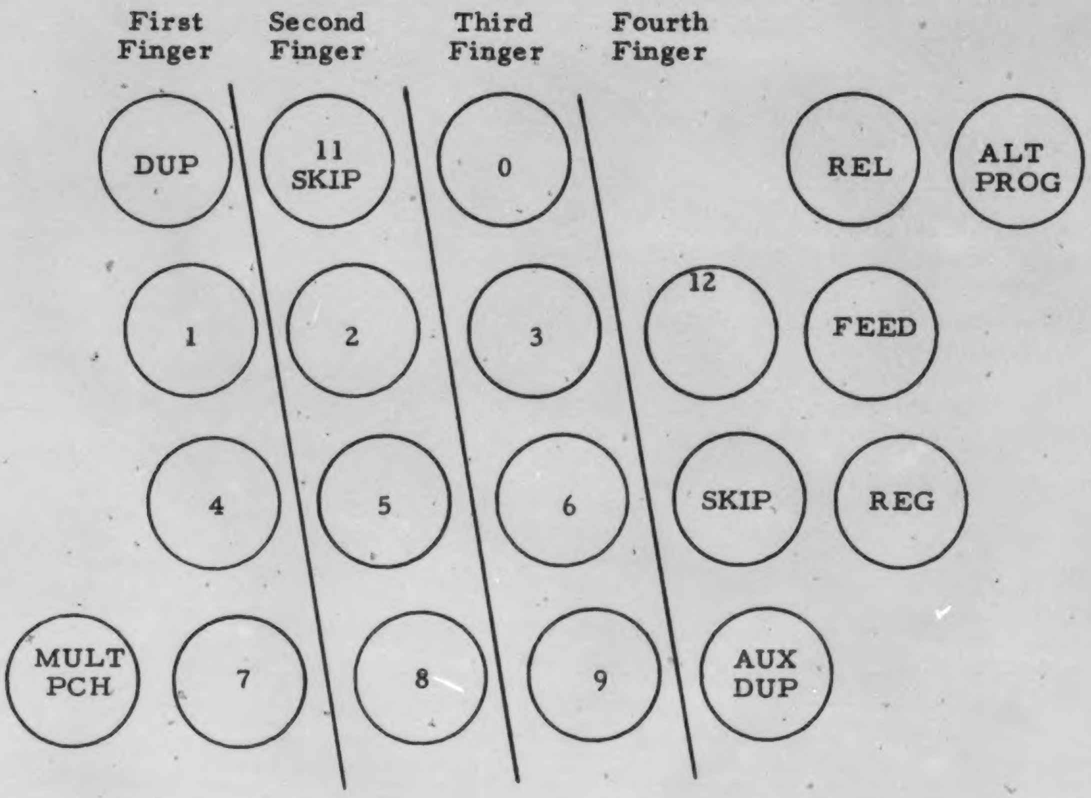
11. If you are going home, after you have cleared your machine put the punched cards in the salvage box on the teacher's desk (or wherever the teacher says) and the unpunched cards back in the box they came from. Please do not mix them together. Leave the card hopper in the open position.

12. Be sure the Star Wheels are UP, then remove the program drum from the spindle and remove the program card from the drum. (Turn handle half way, then remove card from under teeth edge of the clamping strip; turn the handle toward the smooth edge of the clamping strip and finish removing the card from the drum.) Save the program card for the next class session. Return the program drum to the spindle loosely. Do not try to fasten the peg in the hole.

13. Empty Chip Box (if necessary) and replace in machine.

14. Replace books and rulers (if borrowed) back where they came from.

D-9


First Finger
Second Finger
Third Finger
Fourth Finger
DUP
11 SKIP
0
REL
ALT PROG
1
2
3
12
FEED
4
5
6
SKIP
REG
MULT PCH
7
8
9
AUX DUP

NORMAL PROGRAM CODES

12 - FIELD DEFINITION
11 - SKIPS
0 - DUPLICATES
1 - ALPHABETIC
Space - NUMERIC

The 12 zone punch is made by punching the letter "P" with the numeric key held down.

The 11 zone punch is made by punching the SKIP X or DASH SKIP key in either the numeric or alphabetic shift.

The 0 (zero) zone punch is made by punching the 0 (zero) key with the numeric key held down.

12 is the FIELD DEFINITION which continues and defines fields.

RULES:

1. There must be 12's punched in every column of the field except the first position, in the first position there must be the command.
2. Alphabetic must have 1's in every column of the field plus Rule #1.

1. SKIP: SKIP X (11 zone punch) followed by 12's.
2. DUPLICATION: (Numeric) 0 (zero) followed by 12's.
3. DUPLICATION: (Alphabetic) 0-1 (zero and 1 in the same column) followed by the letter "A".**
4. ALPHABETIC: 1 followed by "A's".**
5. NUMERIC: Space followed by 12's.

ALL ALPHABETIC CODING MUST BE FOLLOWED BY A's.

ALL NUMERIC CODING MUST BE FOLLOWED BY 12's.

**The letter "A" is just a faster way of punching a 12 zone punch plus a 1 for an alphabetic field definition. (The letter "A" being made of the zone 12 and the digit 1.)

APPENDIX E

Laboratory Exercises - Keypunch Course

EXERCISE 1

Specification: Card Column

1–6	Student number (last column will be alpha)
7–31	Course title (alpha)
32–35	Course number
36–40	Class number
41–44	Department number (dup)
45	Hours
46–69	Student name (alpha)
78–79	Grade number
80	Student code (alpha)

1–6	7–31	32–35	36–40	41–44	45	46–69	78–79	80
Student Number	Course Title	Course Number	Class Number	Dept. Number	Hour	Student Name	Grade Number	Student Code
87906A	DATA PROCESS.	7634	97531	1068	7	JOHNSON, WILLIAM	81	J
75689B	BOOKKEEPING	5704	78654	3542	7	SMITH, MARYANN	64	S
83824H	BUSINESS	9874	34867	2033	8	ARNOLD, PAUL	40	A
70890N	KEY PUNCH	8756	12308	4286	9	BRIGHT, BETTY	76	B
68570C	HISTORY	9049	67491	0798	7	CARLTON, THOMAS	56	C
12367X	ENGLISH	7878	34786	0012	9	CONNER, FRED	87	C
70956K	SPEECH	6757	23498	7070	9	ISAAC, JACK	13	I
09876T	BUSINESS MATH	7856	45365	4563	8	IRWIN, STANLEY	45	I
75634R	MACHINES	9850	36759	6875	9	KRAVITSKY, JULIUS	68	K
45891W	DRAWING II	8762	54540	2309	7	ROONEY, HARRY	91	R

EXERCISE 2

Specification: Card Column

1–8	Tag number (last column will be alpha)
9–13	Account number
18–22	Quantity
23–30	Amount
45–50	Date (dup)
51–59	Purchase order
60–76	Vendor's name
77–80	Current date (month and day)

1–8	9–13	18–22	23–30	45–50	51–59	60–76	77–80
Tag Number	Account Number	Quantity	Amount	Date	Purchase Order	Vendor's Name	Current Date
7860943A	75972	00024	000540.70	091070	976511578	WILFRED, GODFREY	0812
9865002T	87307	00300	000900.78	091070	976110054	ANDREW, JOHN	0905
7594608R	52119	00019	000067.00	091070	111453379	MONSEN, ARVID	0906
7945009K	67511	00210	000345.90	091070	989966001	SMITH, ROBERT	1109
5674388P	56731	00054	000104.00	091070	768490012	INGRAHAM, FRANK	1110
7645998S	89898	00105	000200.00	091070	674589021	COWAN, KENNETH	0711
2347895D	54761	00078	000175.85	091070	987667893	CLONEY, WILLIAM	1115
5670956J	11223	00230	000414.00	091070	090657411	JOHNSON, FRANKLIN	1205
0987890Z	56098	00025	000050.00	091070	987511004	SEYBOLT, GEORGE	1207
7654562L	87613	00415	000090.98	091070	007423579	WILLIAMS, PETER	1230

EXERCISE 3

Specification: Card Column

1–6	Check number—left zero's print
7–26	Name
27–37	City
38–42	State (duplicate)
50–55	Date (duplicate month and year only)
65–69	Amount

1–6	7–26	27–37	38–42	50–55	65–69	70–80
Check Number	**Name**	**City**	**State**	**Date**	**Amount**	**X**
7589	FREDERICK, ROBERT	LAKEWOOD	CALIF	041067	56.00	
864	MEEKS, BOB	LONG BEACH	CALIF	041467	145.00	
1090	MYERS, FLOYD	BELLFLOWER	CALIF	041667	70.00	
45	HODGE, PAUL	PARAMOUNT	CALIF	042067	35.00	
490	GOFF, DARRELL	DOWNEY	CALIF	042167	265.00	
24	GUIDA, DANNY	LAKEWOOD	CALIF	042567	9.98	
18	BARTLEY, EDWARD	NORWALK	CALIF	042867	80.80	
6785	DEMPSEY, HERBERT	LONG BEACH	CALIF	042967	75.25	
98401	LEONI, ALLEN	ORANGE	CALIF	043067	123.89	
690	HALLIDAY, LINDA	PARAMOUNT	CALIF	043067	300.78	

EXERCISE 4

Specification: Card Column

1–6	Check date (duplicate year only)
12–15	Check number
16–40	Name of payee
46–50	Amount
55–70	Bank name
75–80	Current date (duplicate year only)

1–6	12–15	16–40	46–50	55–70	75–80
Check Date	**Check Number**	**Name of Payee**	**Amount**	**Bank Name**	**Current Date**
030967	2356	HANSON, SUSAN	087.90	FIRST NATIONAL	061068
031167	7645	SCOTT, SYLVIA	090.00	FEDERAL SAVINGS	061868
042167	9856	GARRISON, KENNETH	145.40	FIRST WESTERN	092568
050767	4523	ROONES, BOB	175.00	FIRST NATIONAL	092868
050967	6745	POTTER, TOM	230.50	FIRST WESTERN	100268
072367	8007	HUGHES, MARTIN	16.00	FEDERAL SAVINGS	100668
072467	8009	ANDERSON, EDWIN	43.87	HOME SAVINGS	102468
100967	6731	HARTLEY, DEBBIE	66.65	FIDELITY	111968
102467	1816	HIGGINS, HENRY	303.00	BANK AMERICA	112068
113067	8611	MCCORMICK, JOSEPH	212.20	SAN FRANCISCO	122968

EXERCISE 5

Specification: Card Column

1–3	School number
10–25	Location (alpha)
27–35	Serial number
38–45	Date
55–61	Phone number
65–69	Invoice number
73–75	Items (numeric)
77–79	Item by code (alpha)

1–3	10–25	27–35	38–45	55–61	65–69	73–75	77–79	80
School Number	Location	Serial Number	Date	Phone Number	Invoice Number	Item	Item Code	X
789	DE MILLE	415456120	06-15-65	9276096	36241	104	ABC	
586	LINDBERGH	786540978	06-23-65	8679775	83620	26	BCF	
411	MILLIKAN	456119876	06-25-65	4214580	43689	3	GAB	
871	LAKEWOOD	456681189	07-04-66	4267589	16231	264	LAC	
830	AVALON	564789125	07-19-66	8697460	98743	59	DEF	
764	BANCROFT	761190856	10-20-66	8675980	28623	300	BOG	
597	LONGFELLOW	546781190	11-25-66	4386549	46809	150	LBR	
543	JORDON	567489076	08-06-67	8769458	67540	756	KLH	
984	WILSON	765234901	08-11-67	8675409	52110	98	IJK	
974	POLY	768905437	09-28-67	4257890	87615	9	THR	

EXERCISE 6

Specification: Card Column

1–8	Number Column 8 alpha Columns 1–7 left zero print
14–18	Account number
23–27	Quantity
33–38	Amount (print suppress amount)
46–51	Current date (dup)
52–70	Name
78–80	Your initials (first, middle, last)

1–8	14–18	23–27	33–38	46–51	52–70	78–80
Number	Account Number	Quantity	Amount	Current Date	Name	Initials
98745K	76543	00342	98.00	081567	COTTLE, JAMES	JMC
4590D	98561	00076	180.90	081567	HOLLEMAN, RONALD	RDH
47A	67450	00211	79.00	081567	WHITE, BILL	BIW
0089674J	65231	00098	240.00	081567	PACKARD, JEFF	JAP
5118G	23211	00460	90.00	081567	JAMES, RICHARD	RAJ
78112C	08614	00009	376.98	081567	ZIMMERDAHL, STEVE	SPZ
89T	45367	00075	211.00	081567	WITT, DIANE	DMW
6173M	98457	00250	65.50	081567	MCMURRAY, CHUCK	CRM
7171000F	56897	00780	44.00	081567	COOPER, SHARON	SFC
6591P	11994	00065	300.00	081567	VALLES, DALE	DTV

THE JOB INSTRUCTION SHEET: EXERCISE 1

The following exercise is designed to familiarize the student with the job instruction sheet. Program cards are to be coded and punched from the information contained on the instruction sheet. The prepared program cards are then to be utilized in the punching of the exercise.

The numeric portions of the exercise may be used as an additional numeric drill on the Selectric typewriter.

General Instructions: 24, 26, and 29 Keypunches

1. Using a blank card and a pencil, design a program card to punch the data specifications contained on the job instruction sheet.
2. From the planning card prepared from step 1, punch the program card.
3. Keypunch each page of the exercise without stopping. Time yourself on each page so that you may check your speed.
4. Visually verify the accuracy of your cards, or verify them on a 56 or 59 verifier.
5. Rekeypunch the cards containing errors and turn in all cards to the instructor.
6. Record your time and speed from the accompanying chart in your notebook and also on a separate card to be turned in to your instructor.

Selectric Typewriter Instructions

1. Set margin stops at 1 and 80.
2. Place a blank sheet of paper in the typewriter and be sure the 1428 typing element is in place.
3. Set the tab stops to tabulate over alphabetic fields or fields designed to be skipped.
4. Type the exercise. Upon completion of each line visually verify and make any necessary corrections by retyping the entire line.
5. Time yourself and retype the assignment going straight through and not stopping after each line to verify.
6. Use the accompanying chart to find your punching speed measured in keystrokes per hour. Record the figures in your notebook and also at the bottom of the assignment to be turned in to your instructor.
7. Visually verify the exercises and retype any line containing an error.

Estimating Your Keypunching Speed

Number of Keystrokes Per Page of Exercise	Time Required to Keypunch One Page of Exercise in Minutes	Keystroke Speed in Keystrokes Per Hour
1,040	21	3,000
	15½	4,000
	12½	5,000
	10½	6,000
	7½	8,000
	6	10,000
	5	12,000

Job PER–1

Plant and Equipment Record

Location				Item Number	Asset			Description	Acquisition Date	Expiration Date	Total Installed Cost	Periodic Depreciation	Factor Value	X
Plant	Bldg.	Floor	Dept.		Acct.	Class	Type							
1	1	1	5	4725	5	1	123	AUTO SCREW MACHINE	06/51	06/67	3,473.60	201.24	3,775.46	
1	1	1	5	4462	5	2	125	DRILL PRESS MODEL 2	04/50	04/66	3,600.00	313.00	3,671.00	
1	1	1	5	4462A	5	2	125	SAFETY DEVICE	04/50	04/66	96.00	6.00	98.00	
1	1	1	5	4639	5	3	015	ARBOR PRESS	10/70	10/70	595.05	29.40	619.55	
1	1	1	5	4706	5	3	39	DUPLIGRAPH PRESS	04/51	04/67	2,375.19	145.20	2,568.75	
1	1	1	5	4672	5	5	10	ROTARY SURFACE GRINDER	12/50	12/65	4,600.25	370.44	4,970.69	
1	1	1	5	4680	5	5	16	RADIAL GRINDER	01/51	01/66	1,015.26	66.24	1,087.02	
1	1	1	5	4632	5	5	39	SURFACE GRINDER	10/50	10/65	3,716.44	168.48	2,856.84	
1	1	1	5	4690	5	6	15	MILLING MACHINE	02/51	02/67	2,858.49	168.00	3,054.49	
1	1	1	5	4695	5	6	22	VER. MILLING MACH.	02/51	02/67	3,387.38	207.60	3,629.58	
1	1	1	5	4686	5	6	29	GEAR HOBBING MACH.	02/51	02/67	3,387.38	150.00	2,913.18	
1	1	1	5	4589	5	6	102	GEAR CUTTING MACHINE	08/50	08/68	3,851.59	204.00	3,987.59	
1	1	1	5	4501	6	2	18	PRECISION GAUGE	06/50	06/60	302.40	29.28	317.04	
1	1	1	5	4601	7	1	25	LATHE BENCH	08/50	08/60	482.35	48.24	514.51	
1	1	1	5	4801	7	1	28	LATHE BENCH	08/51	08/67	1,318.50	75.00	1,443.50	
1	1	1	5	4509	7	1	46	TOOL RACK	06/50	06/70	124.80	6.24	127.92	

Job PER–2

Plant and Equipment Record

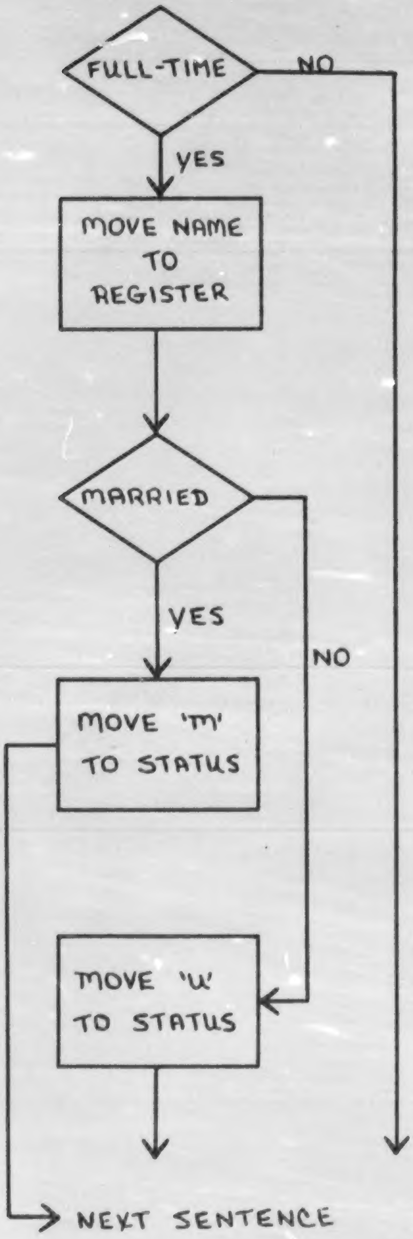
Location				Item Number	Asset			Description	Acquisition Date	Expiration Date	Total Installed Cost	Periodic Depreciation	Factor Value	X
Plant	Bldg.	Floor	Dept.		Acct.	Class	Type							
7	1	7	4	5646	7	2	185	ENGINE LATHE	03/67	03/77	7,090.00	430.00	7,100.00	
7	1	7	4	7773	8	2	846	BENCH MILL	05/67	05/77	9,800.00	650.00	10,050.00	
7	1	7	4	7659	8	2	88	TAPS & DIES	05/67	05/76	650.00	125.00	700.00	
7	1	7	4	4563D	8	2	410	END-MILLS	02/68	02/78	8,000.00	590.55	8,259.90	
7	1	7	4	6530	8	2	550	MILL-CUTTER	02/67	02/77	6,550.00	400.90	7,000.00	
7	1	7	4	1174	8	2	218	REOMERS	07/67	07/77	900.00	220.00	1,100.45	
7	1	7	4	3468	8	2	90	CARBIDE	08/67	08/78	1,500.00	333.54	1,870.00	
7	1	7	4	5587	4	2	56	KELLOGG SPRAY	08/66	08/76	475.00	98.00	550.89	
7	1	7	4	5171	1	2	679	HARDING HEL. CHUCKER	08/67	08/77	8,050.50	600.00	9,100.00	
7	1	7	4	4411A	1	5	643	AIR COMPRESSOR	10/67	10/77	500.00	100.00	650.00	
7	1	7	4	7639	1	5	992	LINCOLN WELDER	10/66	10/77	2,000.00	350.98	2,500.00	
7	1	7	4	6743	1	2	594	SOCKET SET	10/67	10/77	150.00	25.00	175.00	
7	1	7	4	8653	7	2	19	GEAR HOBBING MACH.	09/68	09/78	3,337.38	388.00	4,000.00	
7	1	7	4	6753K	7	5	640	RADIAL GRINDER	11/67	11/76	1,090.45	325.68	2,329.00	
7	1	7	4	6754	8	5	81	ARBOR PRESS	11/67	11/76	635.00	186.00	760.00	
7	1	7	4	9899	8	5	403	BELT SANDER	12/67	12/77	400.00	94.56	575.00	

APPENDIX F

Preliminary Exercises - Business Data Processing Course

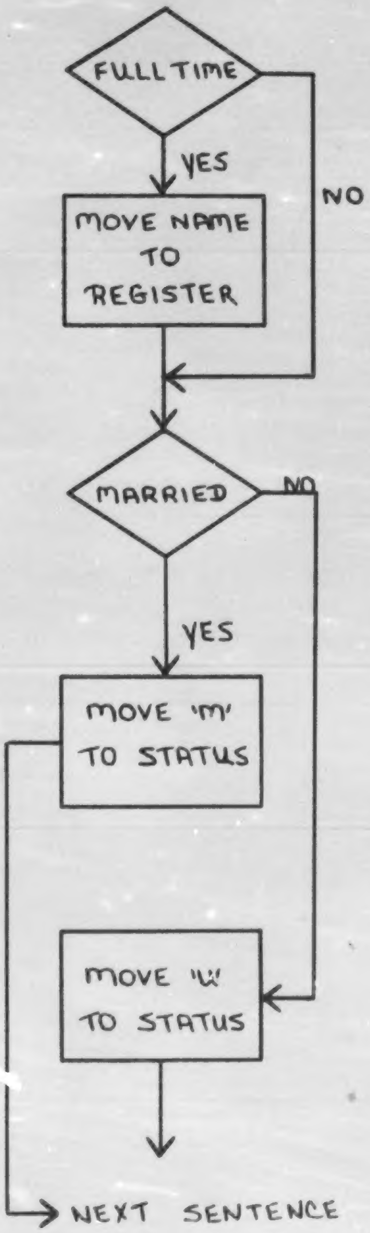
SECTION A

IF FULL-TIME MOVE NAME TO REGISTER. IF MARRIED MOVE @M@ TO STATUS, ELSE MOVE @U@ TO STATUS.


FULL-TIME
NO
YES
MOVE NAME TO REGISTER
MARRIED
YES
NO
MOVE 'M' TO STATUS
MOVE 'U' TO STATUS
NEXT SENTENCE

-NESTED- IF

IF FULL-TIME MOVE NAME TO REGISTER. IF MARRIED MOVE @M@ TO STATUS, ELSE MOVE @U@ TO STATUS.


FULL TIME
YES
NO
MOVE NAME TO REGISTER
MARRIED
NO
YES
MOVE 'M' TO STATUS
MOVE 'U' TO STATUS
NEXT SENTENCE

-NOT NESTED-

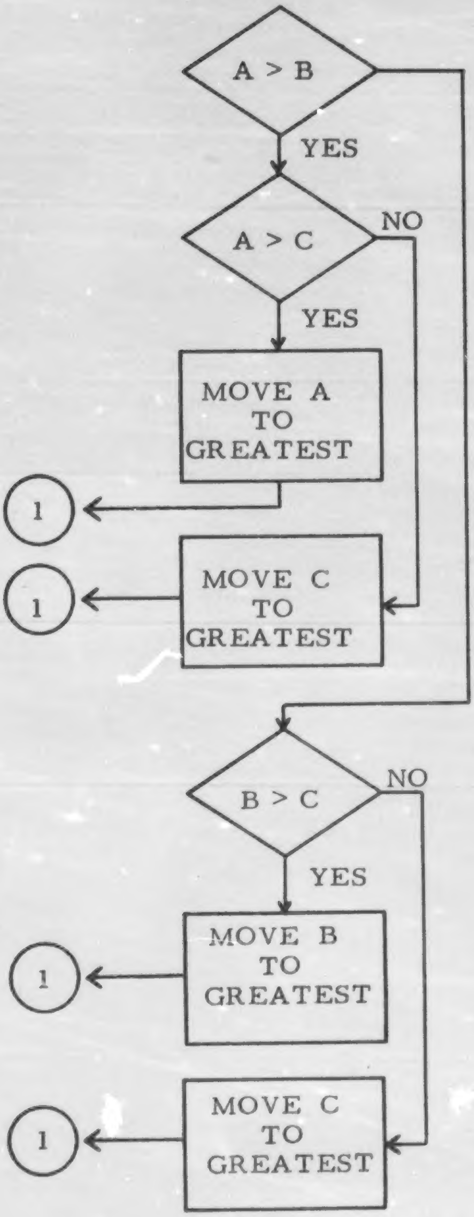
SECTION A

ROUTINE TO FIND LARGEST OF THREE NUMBERS USING NESTED IF's.

IF A > B, IF A > C MOVE A TO GREATEST; ELSE MOVE C TO GREATEST.

IF B > C MOVE B TO GREATEST; ELSE MOVE C TO GREATEST.

FLOW CHART

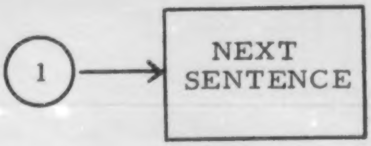

A > B
YES
A > C
NO
YES
MOVE A TO GREATEST
1
MOVE C TO GREATEST
1
B > C
NO
YES
MOVE B TO GREATEST
1
MOVE C TO GREATEST
1

CODING:

```
  IF A > B GO TO PAR-1.
  IF B > C MOVE B TO GREATEST;
  ELSE MOVE C TO GREATEST.
  GO TO PAR-2.

PAR-1.
  IF A > C MOVE A TO GREATEST;
  ELSE MOVE C TO GREATEST.

PAR-2.
  NEXT SENTENCE ...
```


1
NEXT SENTENCE

8. WRITE THE STATEMENTS TO DO THE FOLLOWING:

   A. DETERMINE WHETHER HOURS-WORKED IS GREATER THAN 37.5 AND, IF SO, TRANSFER CONTROL TO ANOTHER PARAGRAPH (WHICH YOU DO NOT WRITE) NAMED OVERTIME-ROUTINE.

   IF HOURS-WORKED IS GREATER THAN 37.5 GO TO OVERTIME-ROUTINE.

   B. TRANSFER CONTROL TO BAD-CODE IF CODE CONTAINS ANYTHING BUT DIGITS.

   IF CODE IS NOT NUMERIC GO TO BAD-CODE.

	SOURCE AREA		RECEIVING AREA	
	PICTURE	SAMPLE DATA	PICTURE	EDITED RESULT.
9.	9(6)	000123	ZZZ,999	123
10.	999999	000008	ZZZ,999	008
11.	9999V99	001234	$$,$$9.99	$12.34
12.	X(6)	123456	XXXBBBXXX	123000456

WRITE THE FIRST THREE LINES OF OUTPUT, GIVEN:

```
IDENTIFICATION DIVISION.
              :
WORKING-STORAGE SECTION.
77   TAX-RATE                  PICTURE V999 VALUE .045.
77   MID-VAL                   PICTURE 9V999.
77   TAX-AMOUNT                PICTURE V99.
77   TAX-CTR                   PICTURE V99.
77   XXX                       PICTURE 9V99.
01   LINE-IMAGE
     02   LOW-1                PICTURE Z.99.
     02   FILLER               PICTURE X(4) VALUE SPACES.
     02   HIGH-1               PICTURE Z.99.
     02   FILLER               PICTURE X(4).
     02   TAX                  PICTURE .99.
PROCEDURE DIVISION.
P-1.
     DISPLAY @   AMOUNT        TAX@.
     DIVIDE TAX-RATE INTO 0.0047 GIVING MID-VAL, ROUNDED.
     MULTIPLY MID-VAL BY 2.0 GIVING XXX.
     MOVE ZERO TO LOW-1.
     MOVE MID-VAL TO HIGH-1.
P-2.
     DISPLAY LINE-IMAGE.
     IF MID-VAL IS GREATER THAN 2 STOP @JOB@.
     ADD .01 TO MID-VAL
     MOVE MID-VAL TO LOW-1
     ADD XXX TO MID-VAL MOVE MID-VAL TO HIGH-1
     ADD .01 TO TAX-CTR MOVE TAX-CTR TO TAX.
     GO TO P-2.
```

```
AMOUNT                 TAX
 0        .10          000
.11       .31          .01
.32       .52          .02
        (OUTPUT)
```

SECTION B

COBOL Word

Of the total COBOL character set, only the alphabetic, numeric, and one special character (the hyphen) are used in the formation of COBOL words. A COBOL word is ended by either a space, or another punctuation symbol.

Data-Names

A data-name is a COBOL word which the programmer invents to represent the data involved in his problem. A data-name may have from one to thirty characters, but neither the first nor the last character may be a hyphen, and at least one of the characters must be alphabetic.

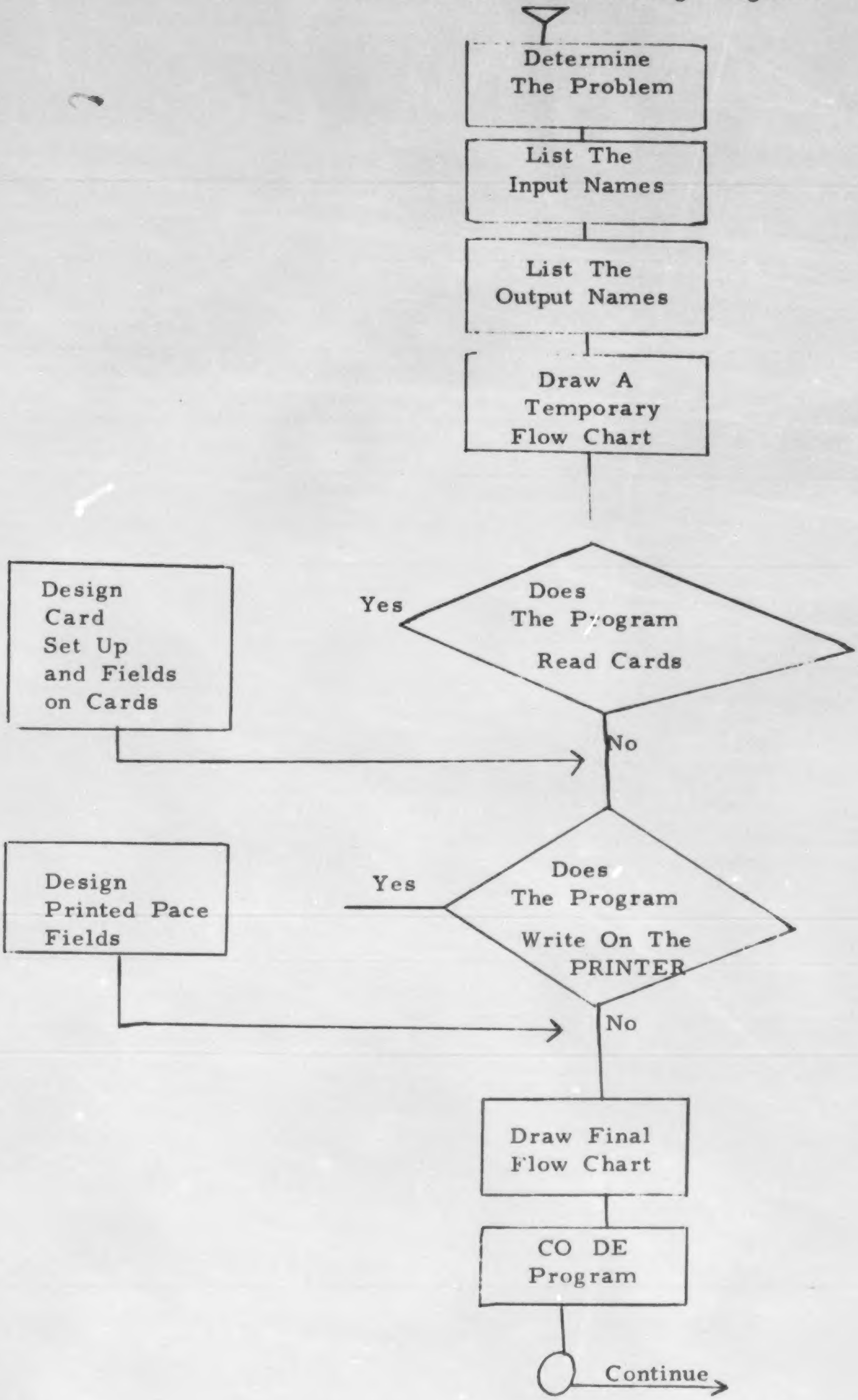
ADD, SUBTRACT, MULTIPLY, DIVIDE, MOVE STATEMENTS

```
ADD       A B C D E F   GIVING X.
SUBTRACT  A B K L J R   FROM  TOTAL  GIVING  LEFT.
MULTIPLY  A  BY  B      GIVING  C.
DIVIDE    A  INTO  B    GIVING  C.
MOVE      B  TO  C
```

Explain Reserve Words

SECTION B

Method to Use When Writing Programs


Determine The Problem
List The Input Names
List The Output Names
Draw A Temporary Flow Chart
Design Card Set Up and Fields on Cards
Yes
Does The Program Read Cards
No
Design Printed Pace Fields
Yes
Does The Program Write On The PRINTER
No
Draw Final Flow Chart
CO DE Program
Continue

SECTION B

```
DATA DIVISION,
WORKING-STORAGE SECTION.
01      EMPL-NAM1     PICTURE    x(3) ,
01      EMPL-NAM2     PICTURE    x(9) ,
01      EMPL-NAM3     PICTURE    x(6) ,
01      EMPL-NAM4     PICTURE    x(9) ,
01      EMPL-NAM5     PICTURE    x(6) ,
01      EMPL-NAM6     PICTURE    x(30) .
PROCEDURE DIVISION,
ST.     MOVE @ ROBIN@        TO    EMPL-NAM1 .
        MOVE @ RUTHΔ GARCIA@    TO        EMPL-NAM2 ,
        MOVE @ WILLIAM@      TO    EMPL-NAM5 .
        MOVE @ IRMA@         TO    EMPL-NAM3 ,
        MOVE @ IRENEΔ CHAVEZ@   TO        EMPL-NAM4 .
        MOVE @ LEILANIΔ MORRIS@ TO        EMPL-NAM6 .
        DISPLAY          EMPL-NAM1.
        DISPLAY          EMPL-NAM2 .
        DISPLAY          EMPL-NAM3 .
        DISPLAY          EMPL-NAM6 .
        DISPLAY          EMPL-NAM5 .
        DISPLAY          EMPL-NAM4 .
        STOP RUN .
```

1). After execution of the program describe the following memory locations:

EMPL-NAM1 ________________ EMPL-NAM2 ________________

EMPL-NAM3 ________________ EMPL-NAM4 ________________

EMPL-NAM5 ________________ EMPL-NAM6 ________________

2). What has been printed on the printer?

3). What would have occurred if all of the pictures in the Data Section were x(14)?

SECTION B

```
DATA DIVISION
WORKING-STORAGE SECTION.
01      COUNT              PICTURE    9999v999 .
01      INT                PICTURE       9(5)   .
01      CTT                PICTURE       9v99   .
01      OWP                PICTURE       999v9 .
01      IOU                PICTURE        99.
01      TZE                PICTURE           99999v9999.
01      PQK                PICTURE       99v999.
01      STU                  PICTURE        v99.
01      AD1                PICTURE    x(6) .
PROCEDURE DIVISION.
SA.     MOVE               213.578    TO       COUNT,
        MOVE               213.578    TO       CTT.
        MOVE               213.578    TO       INT.
        MOVE               213.578    TO       OWD.
        MOVE               213.578    TO       IOU.
        MOVE               213.578    TO       TZE.
        MOVE               213.578    TO       PQR.
        MOVE               213.578    TO       STU .
        MOVE     @         ABCDEFG@   TO       AD1 .
        DISPLAY            INT.
        DISPLAY            TZE.
        DISPLAY            STU.
        STOP               RUN .
```

1). After execution of the program describe the following memory locations:

COUNT ______ INT ______ CTT ______

OWD ______ IOU ______ TZE ______ STU ______

PQR ______ AD1 ______ .

2). What has been printed in the printer?

3). What would have occurred if all pictures were 9999v9999?

SECTION B

#1 IF A IS GREATER THAN B MOVE 5 TO C
MOVE 3 TO D MOVE 8 TO E. MOVE 7 TO X.

	CASE I			CASE II			CASE III		
BEFORE	A 7	B 9	C 14	A 4	B 1	C 14	A 2	B 2	C 14
	D 4	E 5	X 11	D 4	E 5	X 11	D 4	E 5	X 11
AFTER	A	B	C	A	B	C	A	B	C
	D	E	X	D	E	X	D	E	X

#2 IF A IS GREATER THAN B MOVE 5 TO C
MOVE 3 TO D. MOVE 8 TO E. MOVE 7 TO X.

	CASE I			CASE II			CASE III		
BEFORE	A 7	B 9	C 14	A 4	B 1	C 14	A 2	B 2	C 14
	D 4	E 5	X 11	D 4	E 5	X 11	D 4	E 5	X 11
	A	B	C	A	B	C	A	B	C
	D	E	X	D	E	X	D	E	X

#3 IF A IS GREATER THAN B MOVE 5 TO C.
MOVE 3 TO D MOVE 8 TO E. MOVE 7 TO X.

	CASE I			CASE II			CASE III		
BEFORE	A	B	C	A	B	C	A	B	C
	D	E	X	D	E	X	D	E	X
AFTER	A	B	C	A	B	C	A	B	C
	D	E	X	D	E	X	D	E	X

APPENDIX G

Examinations - Business Data Processing Course

SECTION A

1. Hollerith code is described as a/an Code.
   - (a) digital
   - (b) numeric
   - (c) alphanumeric
   - (d) alphabetical

2. The code used on "IBM" punched cards is called:
   - (a) IBM code
   - (b) Hollerith code
   - (c) Morse code
   - (d) None of the above

3. There are (?) columns in the "IBM" punched cards.
   - (a) 60 columns
   - (b) 9 columns
   - (c) 12 columns
   - (d) 80 columns

4. If the GROSS-AMOUNT field of an "IBM" punched card is in columns 36 to 41, inclusive, how many columns does this field contain?
   - (a) 6
   - (b) 5
   - (c) 1
   - (d) None of the above

5. A 12 zone punch and a 5 punch in the same column represents the letter:
   - (a) D
   - (b) N
   - (c) E
   - (d) W

6. (?) holes are necessary to represent special symbols:
   - (a) four
   - (b) three, two, or one
   - (c) two
   - (d) None of the above

7. Card columns assigned to specific items of data are called:
   - (a) unit records
   - (b) fields
   - (c) holenzie kard
   - (d) None of the above

8. (?) multiple punches can be punched in one column of a card:

   (a) 12
   (b) 9
   (c) 3
   (d) None of the above

SECTION A

1. The code used on "IBM" punched cards is called the ____________ code.

2. The program card is locked around the program _______________.

3. The column indicator below the program control unit shows _____ ______________________________.

4. Alphabetic information is recorded in punched cards by combining punches known as the ________________________________ and the _________________________________ punches.

5. An 11 punched in a predetermine card column for control purposes is sometimes referred to as a ________________________ punch.

6. A column or a group of columns set aside to receive specific information is referred to as a ___________________________.

7. Cards prepared from source documents on an IBM Key Punch are checked on a machine known as the __________________________.

8. The printed side of an IBM card is generally referred to as the _____________________________ of the card.

9. There are ____________________ actual punching rows on a card.

10. Special characters (such as the dash, comma, ampersand) are usually made of from 1 to ________ punches in a column.

11. The ____________________________ are raised and lowered by the program-control lever on a key punch machine.

12. Cards are placed in the hopper, ______________________________, _________________________________.

```
11     999999     77777777      000000
11    99999999    77777777     00000000
11    999  999        777      000  000
11     9999999       777       00    00
11        999        77        00    00
11       999         77        000  000
11      999          77        00000000
11     999           77         000000
```

COBOL PROGRAMMING
SECTION A

Write the following arithmetic and standard algebraic notation in COBOL notation:

ARITH-ALGEBRA NOTATION	COBOL NOTATION
1. $a + b$	1. A + B
2. $c - d$	2.
3. xy	3.
4. x^2	4.
5. $z + x - y$	5.
6. $\frac{c}{d}$	6.
7. $b^2 - 4ac$	7.
8. $2(a + b)$	8.
9. $\frac{a + b}{2}$	9.
10. $p + prt$	10.

NUMBERING SYSTEMS

11. ADD: $\begin{array}{r} 10101 \\ \underline{01010} \end{array}$

12. $\begin{array}{r} 1101 \\ \underline{1110} \end{array}$

13. $\begin{array}{r} 1111 \\ \underline{1111} \end{array}$

14. SUBT: $\begin{array}{r} 10101 \\ \underline{00101} \end{array}$

15. $\begin{array}{r} 101011 \\ \underline{011101} \end{array}$

16. $\begin{array}{r} 1101011 \\ \underline{1011110} \end{array}$

Convert the following binary numbers to octal numbers and the resultant octal numbers to their decimal equivalent:

	BINARY NUMBER	OCTAL NUMBER	DECIMAL NUMBER
L	101110	56	46
17.	1001101		
18.	1111111111		
19.	1010000		
20.	001111000		

EOJRTN.

COBOL PROGRAMMING

SECTION A

TRUE OR FALSE

If a statement is ALWAYS TRUE, write (+); if a statement is NOT ALWAYS TRUE, write (0).

____1. A COBOL data name is composed of no more than 30 COBOL characters.

____2. A COBOL statement may contain one and only one sentence.

____3. A COBOL statement must always be followed by a statement separator.

____4. A sentence must always be followed by a period and a space.

____5. Every paragraph must have a name.

____6. The following characters are in the COBOL character set:
+ , % $ # O R 5

____7. The following are DATA NAMES:

NET-PAY
QUANTITY ON HAND
ASTERISK***
DATA-DIVISION
RATE/3

____8. IS is a reserved word.

____9. The following is a legitimate paragraph format:

```
MOVE QUANTITY TO
ORDER-QUANTITY.
MOVE CUSTOMER-NAME TO
REPORT-LINE.
```

____10. The PROCEDURE DIVISION must precede the DATA DIVISION.

Fill in the "after" line in each of the following statements:

11. MOVE A TO B.

Before:	A	B
	165	274
After:	___	___

12. ADD A B C GIVING D.

Before:	A	B	C	D
	1	2	3	4
After:	__	__	__	__

13. SUBTRACT A B C GIVING D.

Before:	A	B	C	D
	2	5	3	8
After:	__	__	__	__

14. MULTIPLY A BY B GIVING C.

Before:	A	B	C
	2	3	4
After:	__	__	__

15. DIVIDE A INTO B GIVING C.

Before:	A	B	C
	3	15	100
After:	__	__	___

16. DIVIDE A B GIVING C ROUNDED.

Before:	A	B	C
	3	11	5
After:	__	___	__

17.

Before:	A	B	C
	6	346	562
After:	__	___	___

18. SUBTRACT A FROM B.

Before:	A	B
	8	2
After:	__	__

19. MULTIPLY X BY Y.

Before:	A	B
	8	009
After:	__	___

20. MULTIPLY UNITS BY PRICE GIVING COST.

Before:	UNITS	PRICE	COST
	23	147	4444444
After:	__	___	_______

SECTION A

1. WHICH OF THESE ARE NOT ACCEPTABLE SEQUENCES OF VERBS, WHEN EACH IS ASSUMED TO APPLY TO ONE FILE?
   - A. READ, READ.
   - B. OPEN, READ, CLOSE.
   - C. OPEN, ACCEPT, CLOSE.
   - D. OPEN, DISPLAY, CLOSE.

2. IN A STATEMENT SUCH AS DISPLAY FINAL-TOTAL UPON PRINTER, WHAT IS THE WORD PRINTER CALLED? HOW IS THIS MADE TO CORRESPOND TO A PHYSICAL DEVICE?

3. CAN A SINGLE DISPLAY BE USED TO OUTPUT BOTH A DATA ITEM AND A LITERAL?

4. CONSIDER THE FOLLOWING SENTENCE:
   - A. DISPLAY @JOB FINISHED.@. HOW MANY PERIODS WOULD BE PRINTED?
   - B. DISPLAY @THIS IS A@, QUOTE, @HORSE@, QUOTE, @.@. EXACTLY WHAT WOULD BE PRINTED?

5. MUST EVERY FILE HAVE A LABEL?

6. TRUE OR FALSE?
   - A. AN OPEN MUST ALWAYS PRECEDE THE FIRST READ OR WRITE FOR A FILE.
   - B. IF THERE IS ONLY ONE READ FOR A FILE, THE AT END "OPTION" MUST BE WRITTEN.

7. WRITE COBOL STATEMENTS TO DO THE FOLLOWING:
   - A. THERE ARE DATA ITEMS IN STORAGE NAMED TRANSACTION-QUANTITY AND QUANTITY-ON-HAND. WRITE A STATEMENT THAT WILL MAKE THE VALUE OF QUANTITY-ON-HAND THE SAME AS THE VALUE OF TRANSACTION-QUANTITY, LEAVING THE LATTER UNCHANGED.
   - B. THERE IS A DATA ITEM NAMED COUNTER: MAKE ITS NUMERICAL VALUE EQUAL TO 1.

C. THERE IS A DATA ITEM NAMED HEADING: MOVE THE CHARACTERS PAGE NUMBER TO IT.

____

D. MAKE THE NUMERICAL VALUE OF THE ITEM NAMED HOW-MANY ZERO.

____

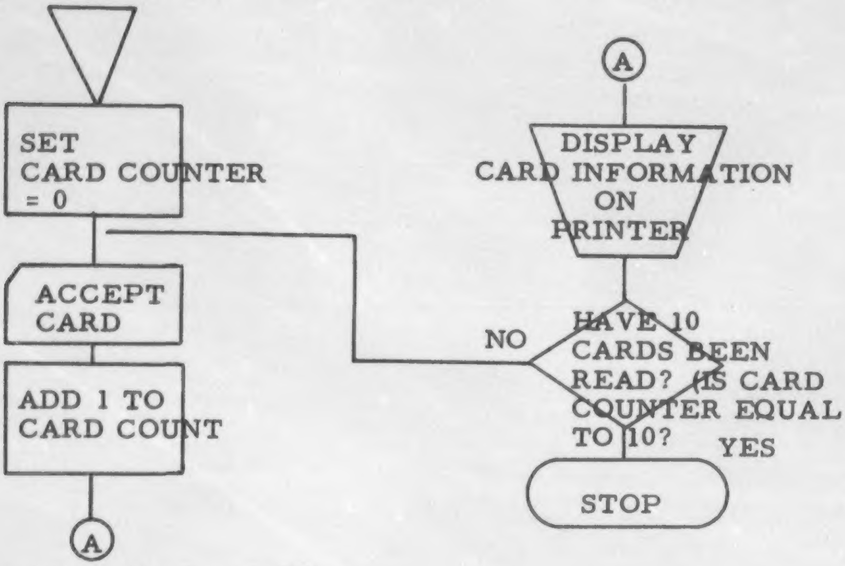
E. DIVIDE THE VALUE OF THE ITEM NAMED TOTAL BY THE VALUE OF THE ITEM NAMED NUMBER, WITH THE QUOTIENT BECOMING THE NEW VALUE OF THE ITEM NAMED AVERAGE.

____

SECTION A

Design a flow chart (block diagram) that represents the reading of 10 cards from the card reader and do the following:

Print the 10 cards in the order read.

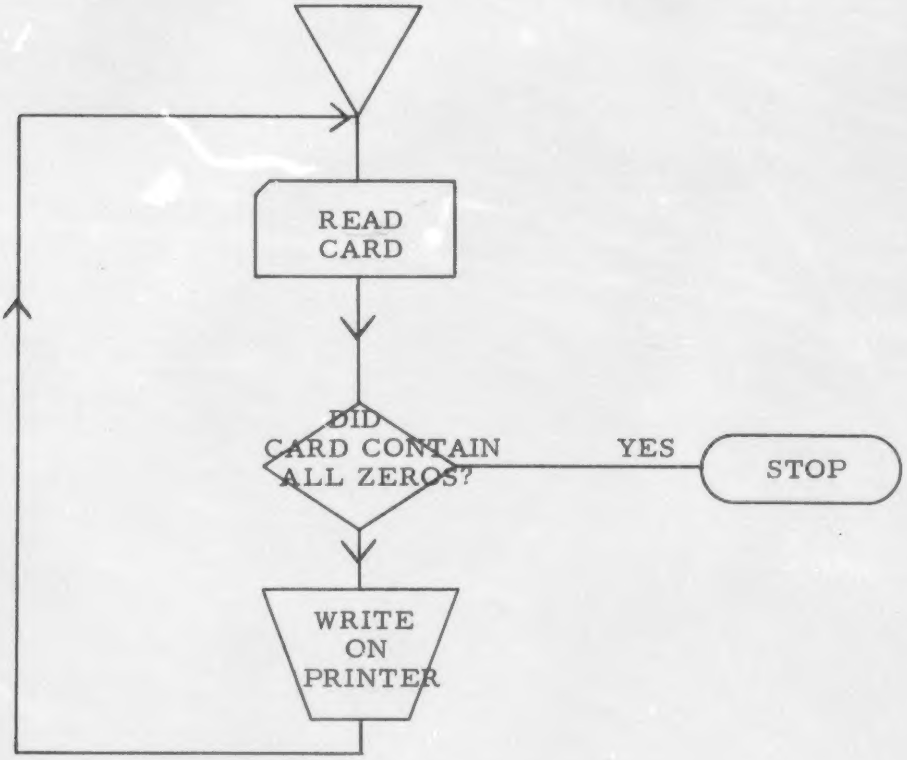

SET
CARD COUNTER
= 0
ACCEPT
CARD
ADD 1 TO
CARD COUNT
A
A
DISPLAY
CARD INFORMATION
ON
PRINTER
NO
HAVE 10
CARDS BEEN
READ? (IS CARD
COUNTER EQUAL
TO 10?
YES
STOP

Only print cards 9 and 10.

Print cards in the reverse order read.

SECTION B

Read an unknown number of cards from the card reader and write them on the printer. It is known that the last card contains all zeros in the reading field and is not to be printed.


READ
CARD
DID
CARD CONTAIN
ALL ZEROS?
YES
STOP
WRITE
ON
PRINTER

SECTION B

Given the following information:

ITEM-NAME Long Eared Lop Nosed Nut
Quantity: 526420
Value-per-unit: 50
Total-value:

ITEM: 14K Plated Lock Washer
Quantity: 25
Value per unit: 5350
Total value: ?

Calculate the total value of the "Long Eared Lop Nosed Nut" and the 14K Plated Lock Washer.

$$\begin{aligned}\text{Total value}_{L,E,L,N,N} &= \text{Quantity} \times \frac{\text{value}}{\text{unit}} = 626420 \times 50\\ &= 526420 \times 50\\ &= 26321000\end{aligned}$$

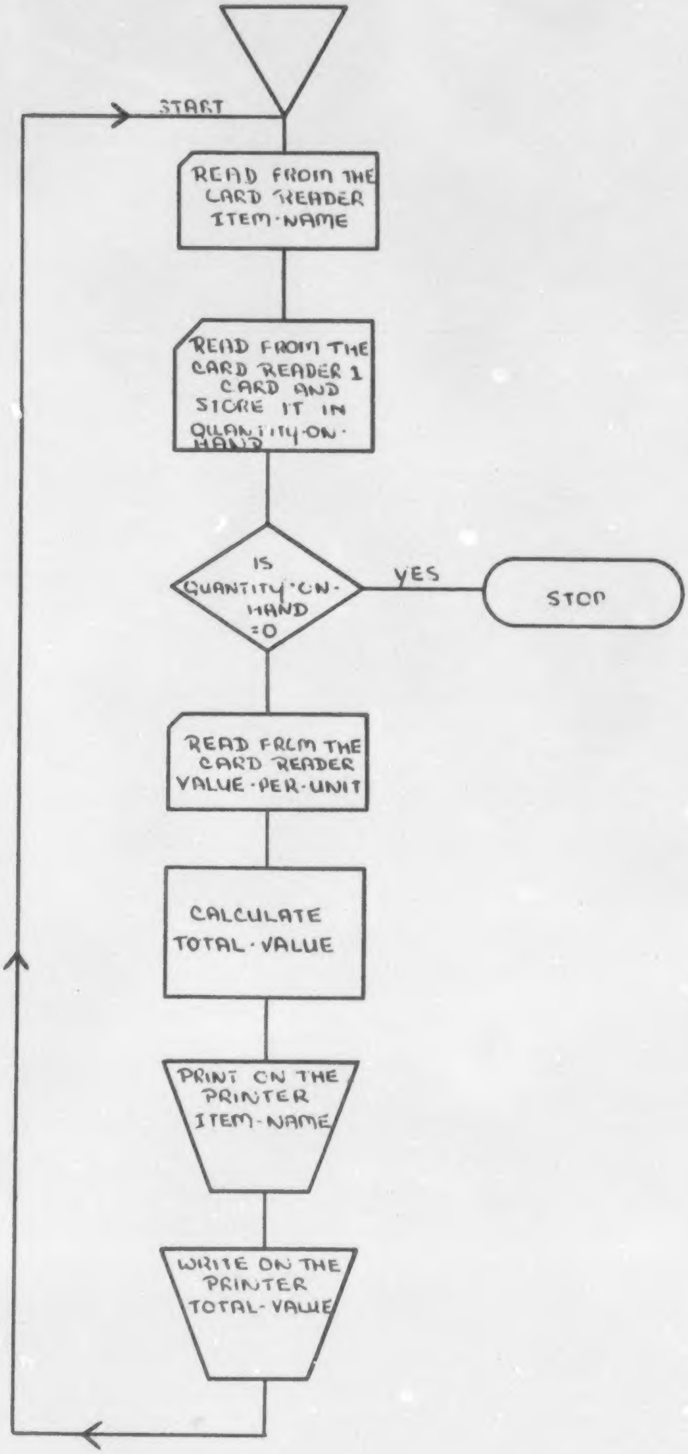
$$\begin{aligned}\text{Total value}_{14K,P,L,W} &= \text{Quantity} \times \frac{\text{value}}{\text{unit}}\\ &= 25 \times 5350\\ &= 133750\end{aligned}$$

Draw a flow chart (block diagram) for a computer program to do the calculations of Part A. (See next page.)

Study the COBOL program on page 127 of Spitzbarth that does these calculations.

Show how the printed results of this program look on the printer (see next page).

B. Flow chart of program on page 127.


START
READ FROM THE CARD READER ITEM-NAME
READ FROM THE CARD READER 1 CARD AND STORE IT IN QUANTITY-ON-HAND
IS QUANTITY-ON-HAND =0
YES
STOP
READ FROM THE CARD READER VALUE-PER-UNIT
CALCULATE TOTAL-VALUE
PRINT ON THE PRINTER ITEM-NAME
WRITE ON THE PRINTER TOTAL-VALUE

D. Output on the printer has the following format:

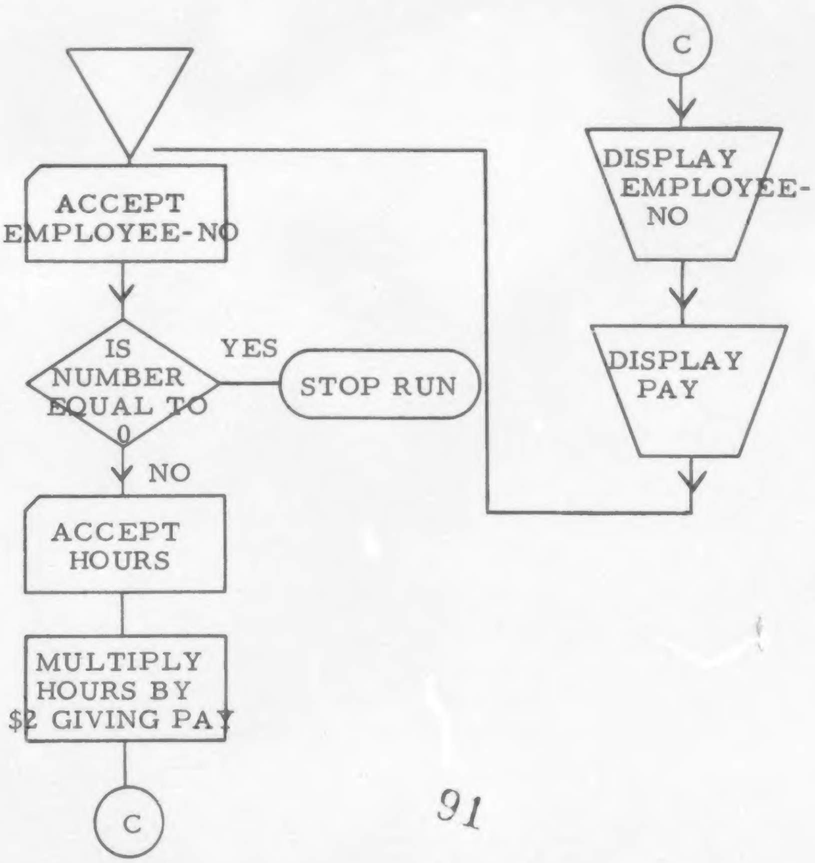
```
         00
         1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24
Line 1   LONG  EARED    L  O  P    N  O  S  E  D     N  U  T
Line 2   2 6 3 2 1 0 0 0
Line 3   1 4K   PLATED    L  O  C  K    W  A  S  H  E  R
Line 4   0 0 1  3 3 7 5 0
```

SECTION B

Compose a flow chart that is designed to calculate the pay of all employees. The employee number and the pay received for the number of hours worked are to be printed on the printer. The employee number and the number of hours worked are read from the card reader in the following order:

Card 1	15293	← Employee No. is 15293
Card 2	40	← No. of hours worked
Card 3	15896	← Employee No. 15896
Card 4	48	No. of hours worked
Card 5	12345	"
Card 6	26	"
		"
		"
Card N	0000	"

The number of employees is unknown, but we do know the last card contains all zeros. Also the rate of pay for all employees is $2 per hour.

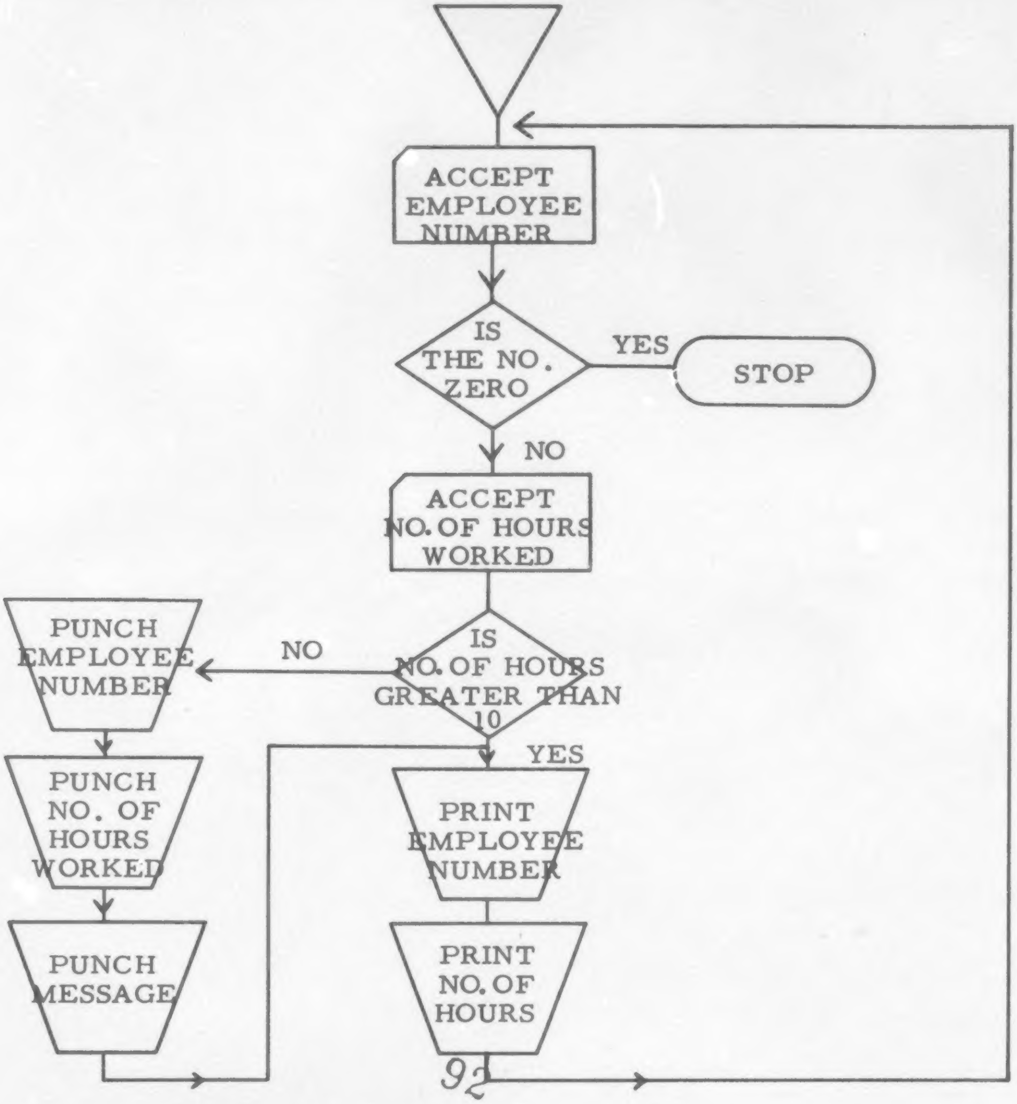

ACCEPT EMPLOYEE-NO
IS NUMBER EQUAL TO 0
YES
STOP RUN
NO
ACCEPT HOURS
MULTIPLY HOURS BY $2 GIVING PAY
C
C
DISPLAY EMPLOYEE-NO
DISPLAY PAY

SECTION B

Design a block diagram (flow chart) to do the following:

Read an unspecified number of cards from the card reader where the odd numbered cards represent an employee number and the even numbered cards represent hours worked. That is, each card containing an employee's employee number is followed by a card containing the number of hours that the corresponding employee worked. The number of cards is unknown, but we have placed a card containing all zeros as the last card.

Display on the printer employee number followed by the number of hours that the employee worked. If an employee worked less than 10 hours, then he has been loafing on the job. Therefore, we also want three cards punched out. The first will contain his employee number. The second will contain the number of hours he worked. The third will be a statement telling him to "earn more money."


ACCEPT EMPLOYEE NUMBER
IS THE NO. ZERO
YES
STOP
NO
ACCEPT NO. OF HOURS WORKED
IS NO. OF HOURS GREATER THAN 10
NO
YES
PUNCH EMPLOYEE NUMBER
PUNCH NO. OF HOURS WORKED
PUNCH MESSAGE
PRINT EMPLOYEE NUMBER
PRINT NO. OF HOURS

SECTION B

Design a card setup and flow chart to do the following:

Print the employee number, employee name, address and salary based upon the number of hours worked for each employee. You can read the employee's number, employee name, address and number of hours worked from the card reader. You know how many employees there are. You know the first 50 employees receive $2 per hour and all the others receive $2.50 per hour.

If you knew how many employees there were each day but the number changes from day to day, then how would you design the card setup and flow chart.

APPENDIX H

COBOL Programming Assignments - Data Processing Course

COBOL PROGRAMMING - CASE STUDY 10

SECTION A

The purpose of this case study is to write a simple payroll program. For the sake of simplicity we will assume we have a master card and a detail card for each employee and that the master card for each employee immediately precedes his detail card. Assume there is no master card without a corresponding detail card and vice versa.

The format for the master card is as follows:

COL.	DESCRIPTION	FORMAT
1-4	EMPLOYEE NO.	9999
	EMPLOYEE NAME	
5-14	LAST	A(10)
15-17	INITIAL	A(3)
18-24	SPACES	
25-26	DEPARTMENT CODE	99
27-28	TAX CLASS	99
29-35	YTD GROSS	99999V99
36-40	YTD FICA	999V99
41-46	YTD WITHHOLDING	9999V99
47-50	HOURLY RATE	99V99
51-52	WEEK	99
53-61	SOCIAL SEC. NO.	9(9)

The format for the detail card is as follows:

1-4	EMPLOYEE NO.	9999
10-11	HOURS WORKED	99
20-21	WEEK	99

To produce a payroll register, the required calculations are as follows:

1. BASE PAY = BASE-REGULAR HOURS X HOURLY RATE.
2. OVERTIME PAY = OVERTIME HOURS X HOURLY RATE X 1.5 WHERE OVERTIME IS ANYTHING OVER 40 HOURS.

3. GROSS PAY = BASE PAY + OVERTIME PAY.

4. WITHHOLDINGTAX = (GROSS PAY = (TAX CLASS X $13.00) X 18%.

5. FICA = 4% OF GROSS PAY IF YTD GROSS IS LESS THAN $4800.

6. NET PAY = GROSS PAY - FICA - WITHHOLDING.

7. NEW YTD GROSS = OLD YTD GROSS + GROSS.

8. NEW YTD FICA = OLD YTD FICA + FICA.

9. NEW YTD WITHHOLDING = OLD YTD WITHHOLDING + WITHHOLDING.

10. THE CALCULATIONS FOR DEPARTMENTAL TOTALS ARE ROUTINE AND WILL NOT BE SPELLED OUT.

SECTION A

OUTPUT:

```
SOC SEC NO     TC   YTDGROSS  DEPT   INDO      NAME        RATE  REG  OT    GROSS     FICA    WITH        NET
XXX XX XXXX    X     XXXX.XX   XX    XXXX  X XXXXXXXXX   X.XX    XX  XX   XXX.XX     X.XX   XX.XX      XXX.XX
XXX XX XXXX    X      XXX.XX   XX    XXXX  X XXXX        X.XX    XX  XX    XX.XX     X.XX    X.XX       XX.XX
      .
      .
      .
      .
XXX XX XXXX    X     XXXX.XX   XX    XXXX  X XXXXX       X.XX    XX  XX   XXX.XX     X.XX   XX.XX      XXX.XX

DEPT TOTALS   XX  $XXXXX.XX                                     XXX  XX  $XXX.XX    $X.XX  $XX.XX   $XXXX.XX
```

Each dollar and cents figures are to be edited by inserting a decimal point and suppressing leading zeros in the department totals.

Start each department's payroll register on a new sheet with the appropriate headings. Due to the amount of space required for the headings, it will be necessary to prepare a careful printers spacing chart.

COBOL PROGRAMMING

SECTION A

EDITING:

	SOURCE AREA		RECEIVING AREA	
	PICTURE	SAMPLE DATA	PICTURE	EDITED RESULT
1.	999999	123456	$ZZZ,ZZZ.99	
2.	9999V99	123456		$bb1,234.56
3.	9(4)V99	000123	$ZZZ,ZZZ.99	
4.	9999V99	000123	$$$$,$$$.99	
5.	999999	000100	$$$,$99.99	
6.	9(6)	000008		$08.00
7.	99999V9	001234	$$$$,9.99	
8.	S9999V99	001234	$*,***.99DB	
9.	S9(6)	123456	-999999	
10.	999999	123456	9B(4)9(5)	

REVIEW:

11. Subtract A B C from D giving X.

Before: 2 5 3 8 4
After: ___ _ _

12. Divide A into B giving C rounded.

Before: 4 19 4
After: _ __ _

13. The nature of CODASYL would be best described by one of the following statements. Select the one.

___a) The group which made the attempt to generate the COBOL language.

___b) The meeting at which the initial specifications for a common business oriented language was devised.

___c) The organization which created COBOL.

14. Match the phrases in Column Y with the related symbols in Column X.

COLUMN X	COLUMN Y
___a) data-name-1	A. Key reserved word
___b) OBJECT-COMPUTER	B. Optional reserved word
___c) MEMORY SIZE	C. Non-reserved words
___d) H-200 HONEYWELL-200	D. Optional COBOL statement
___e) NO SEGMENTATION	E. Choice must be made

15. Check the following literals as either NON for non-numeric and N for numeric.

___a) @12345@

___b) 12345

___c) @12345ABC@

___d) @THIS IS A NUMERIC LITERAL@

___e) +157.567

___f) @03-30-70@

16. Choose the entry in Column Y which properly describes an item in Column X.

COLUMN X	COLUMN Y
___a) DATA DIVISION	A. Provides documentation, program name, etc.
___b) ENVIRONMENT DIVISION	B. Provides equipment configuration and assignments.
___c) PROCEDURE DIVISION	C. Identifies all of the I/O areas.
___d) IDENTIFICATION DIVISION	D. Includes all of the instructions necessary to solve a given problem.

SECTION A

PAGE 312
APPENDIX I
CASE STUDIES

PART I

CASE STUDY # 1	PROBLEM 1	PENCIL PROBLEM
CASE STUDY # 2	PROBLEM 2	COST PROBLEM
CASE STUDY # 3	PROBLEM 5	CONVERSION TABLE
CASE STUDY # 4	PROBLEM 6	SIMPLE INTEREST
CASE STUDY # 5	DITTO	NELLA, ILMO, JAY
EXTRA CREDIT		

PART II

CASE STUDIES TO BE ASSIGNED AT A LATER DATE.

Study Chapter Nine. FILE USAGE Page 185.

PROGRAMMING INFORMATION FOR H-200.

```
IDENTIFICATION DIVISION.
PROGRAM-ID.  PROB-NO.
AUTHOR.   NAME.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER.  H-200-SPECIAL.
OBJECT-COMPUTER.  H-200, SUPERVISOR, NO SEGMENTATION.
SPECIAL-NAMES.
    PAGE IS TO-NEXT-PAGE.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT PUNCHED-CARD-FILE ASSIGN TO CARD-READER E.
    SELECT REPORT-FILE ASSIGN TO PRINTER B.
I-O-CONTROL.
    APPLY H-200-SPECIAL ON PUNCHED-CARD-FILE.
DATA DIVISION.
FILE SECTION.
FD PUNCHED-CARD-FILE,
    LABEL RECORDS ARE STANDARD,
    VALUE OF IDENTIFICATION IS @ @,
    DATA RECORDS IS IN-PUT ...
                  :
                  :
                  :
FD REPORT-FILE LABEL RECORDS ARE OMITTED,
    DATA RECORD IS PRINT-LINE.
                  :
                  :
PROCEDURE DIVISION.
    OPEN INPUT PUNCHED-CARD-FILE, OUTPUT REPORT-FILE ...
```

COBOL PROGRAMMING - CASE STUDY 1, E.C.

SECTION A

The following program is extra credit.

Required - FLOW CHART
CODING SHEET
PRINTING CHART

Write a program to read a deck of cards and print the information on the printer. Skip to the top of a new page before beginning and also if a form overflow occurs.

The format of the cards is as follows:

COL.	INFORMATION	FORMAT
1-5	IDNO	99999
	NAME	
6-20	LAST	A(15)
21-29	FIRST	A(9)
30	MIDDLE-INITIAL	A(1)
31-39	SOC-SECURITY-NO	9(9)
40-41	OCCUPATION-CODE	99
50-54	PAY-RATE	99V999
59-60	DEPARTMENT	99

The format of the printout for each card is as follows:

LINE 1:

10-14	IDNO	99999
18-28	SOC-SECURITY-NO	999-99-9999
35-62	NAME	
	FIRST	A(9)
	MIDDLE-INITIAL	bA.b
	LAST	A(15)

LINE 2:

11-12	OCCUPATION-CODE	99
16-17	DEPARTMENT	99
20-25	PAY-RATE	99.999

Double space between every two line combination: single space within the two line combination. Suppress any leading zeros in pay rate.

Rewrite WRITE IBM-360 program for H-200.

```
IDENTIFICATION DIVISION.
PROGRAM-ID.  'EXTRA CREDIT'
AUTHOR.  TEAM O.
ENVIRONMENT DIVISION.
CONFIGURATION SECTION.
SOURCE-COMPUTER.  IBM-360 E25.
OBJECT-COMPUTER.  IBM-360 E25.
INPUT-OUTPUT SECTION.
FILE-CONTROL.
    SELECT IN-FILE ASSIGN TO 'SYS009' UNIT-RECORD 254OR.
    SELECT OUT-FILE ASSIGN TO 'SYS001' UNIT-RECORD 1403.
I-O-CONTROL.
    APPLY BOTTOM-OF-PAGE TO FORM-OVERFLOW ON OUT-FILE.
DATA DIVISION.
FILE SECTION.
FD  RECORDING MODE IS F.
01  EMP-REC.
    02 IDNO                        PICTURE 99999.
    02 INNAME.
        03 INLAST                  PICTURE A(15).
        03 INFIRST                 PICTURE A(9).
        03 INMI                    PICTURE A(1).
    02 INSSNO.
        03 ONE                     PICTURE 999.
        03 TWO                     PICTURE 99.
        03 THREE                   PICTURE 9999.
    02 INOCCODE                    PICTURE 99.
    02 FILLER                      PICTURE X(8).
    02 INPARATE                    PICTURE 99V999.
    02 FILLER                      PICTURE X(4).
    02 INDEPT                      PICTURE 99.
    02 FILLER                      PICTURE X(20).
FD  OUT-FILE DATA RECORD IS OUT-REC LABEL RECORDS ARE OMITTED
    RECORDING MODE IS F.
01  OUT-REC                        PICTURE X(133).
WORKING-STORAGE SECTION.
01  PRINT-LINE-1.
    02 FILLER                      PICTURE X(6).
    02 ID-OUT                      PICTURE 99999.
    02 FILLER                      PICTURE X(5).
    02 SSOUT.
        03  SSONE                  PICTURE 999.
        03  HYP-1                  PICTURE X(1).
        03  SSTWO                  PICTURE 99.
        03  HYP-2                  PICTURE X(1).
        03  SSTHREE                PICTURE 9999.
    02  FILLER                     PICTURE X( 5).
```

CASE STUDY

SECTION A

Write a program to read a deck of cards and print the information on the printer. Skip to the top of a new page before beginning and also if a form overflow occurs.

The format of the cards is as follows:

COL.	INFORMATION	FORMAT
1-5	IDNO	99999
	NAME	
6-20	LAST	A(15)
21-29	FIRST	A(9)
30	MIDDLE-INITIAL	A(1)
31-39	SOC-SECURITY-NO	9(9)
40-41	OCCUPATION-CODE	99
50-54	PAY-RATE	99V999
59-60	DEPARTMENT	99

The format of the printout for each card is as follows:

LINE 1:

10-14	IDNO	99999
18-28	SOC-SECURITY-NO	999-99-9999
35-62	NAME	
	FIRST	A(9)
	MIDDLE-INITIAL	bA.b
	LAST	A(15)

LINE 2:

11-12	OCCUPATION-CODE	99
16-17	DEPARTMENT	99
20-25	PAY-RATE	99.999

Double space between every two line combination; single space within the two line combination. Suppress any leading zeros in pay rate.

COBOL PROGRAMMING

SECTION A

GIVEN: Input data cards as described in Case Study 5.

REQUIRED: Considering 120 position for a print area, prepare the following report:

```
                    COBOL CASE STUDY 7

            NELLO              ILLMO              JAY
            xxx                xxxx               xxxx
            xxx                xxxx               xxxx
            xxx                xxxx               xxxx
            xxx                xxxx               xxxx
             .                  .                  .
             .                  .                  .
             .                  .                  .
            xxx                xxxx               xxxx
            xxx                xxxx               xxxx

TOTAL      xxxx               xxxxx              xxxxx

AVERAGE     xxx                xxxx               xxxx

(CURRENT DATE)
(FORMAT MMM DD YY)
```

INSTRUCTIONS TO CONSIDER FOR THIS PROBLEM:

READ
WRITE
COMPUTE

SECTION B

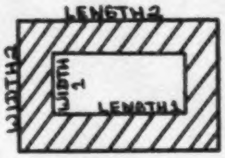
A. Draw a flow chart for each problem.

B. Develop a table of all data-names and show the contents of each data-name after the execution of each instruction.

C. Some problems have additional questions; answer them.

```
       8      12
       START1.
              MOVE 156 TO INT.
(1)           DIVIDE 12 INTO INT GIVING DOZEN-INT.
              ADD 4 DOZEN-INT GIVING DOZEN-INT.
              STOP RUN.
```

```
       8      12
       BEGIN-HERE-AND-NOW.
              MOVE 43 TO HOURS.
              ADD 7 HOURS GIVING BONUS-HOURS.
(2)           MOVE 3.50 TO RATE.
              MULTIPLY HOURS BY RATE GIVING ORDINARY-PAY.
              MULTIPLY BONUS-HOURS BY RATE GIVING BONUS-PAY.
              STOP RUN.
```



```
       8      12
       BEGIN.
              MOVE 5 TO LENGTH1.
              MOVE 13 TO WIDTH1.
              MOVE 20 TO LENGTH2.
              MOVE 14 TO WIDTH2.
              MULTIPLY LENGTH1 BY WIDTH1 GIVING AREA1.
(3)           MULTIPLY LENGTH2 BY WIDTH2 GIVING AREA2.
              SUBTRACT AREA1 FROM AREA2 GIVING SHADED-AREA.
              ADD LENGTH1 LENGTH2 WIDTH1 WIDTH2 GIVING PERIMETER1.
              MULTIPLY LENGTH2 BY 2 GIVING 2TIMES-LENGTH2.
              MULTIPLY WIDTH2 BY 2 GIVING 2-TIMES-WIDTH2.
              ADD 2TIMES-LENGTH2 2-TIMES-WIDTH2 GIVING PERIMETER2.
              STOP RUN.
```

```
        8    12
        OK111.
             MOVE 6 TO NUMBER-OF-CANDIES.
        OK12.
             MOVE .05 TO PRICE-OF-EACH.
             MULTIPLY NUMBER-OF-CANDIES
(4)          BY PRICE-OF-EACH GIVING
             TOTAL-COST-OF-CANDY.
             IF TOTAL-COST-OF-CANDY IS GREATER THAN .25 MOVE 4
             TO NUMBER-OF-CANDIES
             GO TO OK12.
             SUBTRACT .20 FROM TOTAL-COST-OF-CANDY.
             STOP RUN.
```

```
        8    12
        OK11.
             MOVE 6 TO NUMBER-OF-CANDIES.
        OK12.
             MOVE .05 TO PRICE-OF-EACH.
             MULTIPLY NUMBER-OF-CANDIES BY
(5)          PRICE-OF-EACH GIVING TOTAL-COST-OF-CANDY.
             IF TOTAL-COST-OF-CANDY IS GREATER THAN .25 MOVE 4
             TO NUMBER-OF-CANDIES.
             GO TO OK12.
             SUBTRACT .20 FROM TOTAL-COST-OF-CANDY.
             STOP RUN.
```

```
        8    12
        TITLE-ROUTINE.
(6)          MOVE 'EMPLOYEEΔΔNUMBERΔΔPAYΔΔ' TO TITLE.
             DISPLAY TITLE.
             STOP RUN.
```

```
        8    12
        TITLE-ROUTINE.
(7)          DISPLAY 'EMPLOYEEΔΔNUMBERΔΔPAYΔΔ'.
             STOP RUN.
```

```
        8    12
        BEGIN.
             MOVE 5 TO A.
(8)          MOVE 4 TO B.
             ADD A B GIVING C.
             STOP RUN.
```

Find the error(s) in the following programs. (Also do A, B, C)

```
      8      12
      TITLE-ROUTINE.
           DISPLAY 'EMPLOYEEΔΔNUMBER'.
           GO TO XENTLY.
(9)        MOVE 15 TO X.
           MULTIPLY X BY X GIVING X.
      XENTRY.
           STOP RUN.
```

```
      8      12
      HEAD.
           STOP RUN.
(10)       MOVE 5 TO X.
           MULTIPLY X BY 2 GIVING X.
           GO TO HEAD.
```

```
      8      12
      XYENTER.
           GO TO X-START.
      XYZSTARRT.
           MOVE 5 to Y.
(11)       MULTIPLY Y BY W GIVING Y.
           STOP RUN.
      X-START.
           MOVE 2 TO W.
           GO TO XYZSTART.
```

```
      8      12
      XYENTER.
           GO TO X-START.
      XYZSTART.
           MOVE 5 TO Y.
(12)  12   MULTIPLY Y BY W GIVING Y.
           STOP RUN.
      X-START.
           MOVE 2 TO W.
      GO TO XYZSTART.
```

```
      8      12
      BEGIN.
           ADD A B GIVING C.
(13)       MOVE 5 TO A.
           MOVE 4 TO B.
           STOP RUN.
      8     12
      BEGIN.
            MOVE 5 TO A.
(14)        MOVE 4 TO B.
            ADD A B C GIVING C.
            STOP RUN.
```

```
      8     12
      BEGIN.
            STOP RUN.
(15)        MOVE 5 TO A.
            MOVE 4 TO B.
            ADD A B GIVING C.
```

```
      8     12
      DATA DIVISION.
      WORKING-STORAGE SECTION.
      01    INT           PICTURE  99999999
      01    DOZEN-INT   PICTURE   9(8)
      PROCEDURE DIVISION.
(16)  START1.
            ACCEPT INT.
            DIVIDE 12 INTO INT GIVING DOZEN-INT.
            ADD 4 DOZEN-INT GIVING DOZEN-INT.
            DISPLAY DOZEN-INT.
            STOP RUN.
cc
1
00000156  ←  Data Card
```

```
      8     12
      DATA DIVISION.
      WORKING-STORAGE SECTION.
      01    HOURS      PICTURE  99.
      01    RATE       PICTURE  9V99.
      01    BONUS-HOURS       PICTURE  99.
      01    ORDINARY-PAY   PICTURE  999V99.
      01    BONUS-PAY         PICTURE  999V99.
      PROCEDURE DIVISION.
(17)  BEGIN-HERE-AND-NOW.
            MOVE 43 TO HOURS.
            ADD 7 HOURS GIVING BONUS-HOURS.
            MOVE 3.50 TO RATE.
            MULTIPLY HOURS BY RATE GIVING ORDINARY-PAY.
            MULTIPLY BONUS-HOURS BY RATE GIVING
            BONUS-PAY.
            DISPLAY ORDINARY-PAY.
            DISPLAY BONUS-PAY.
            STOP RUN.
```

```
       8   12
       DATA DIVISION.
       WORKING-STORAGE SECTION.
       01  NO1     PICTURE 99
       01  NO2     PICTURE 99
       01  PIN     PICTURE 999
       PROCEDURE DIVISION.
       START.
           MOVE 5 TO NO2.
           MOVE 1 TO NO1.
       OUTPGM.
(18)       IF NO1 IS LESS THAN 25 ADD 5 NO1
           GIVING NO1
           ADD 5 NO2 GIVING NO2
           MULTIPLY NO1 BY NO2 GIVING PIN.
           IF PIN IS LESS THAN 200
           GO TO OUTPGM.
           DISPLAY NO1.
           DISPLAY NO2.
           DISPLAY PIN.
           STOP RUN.
```

Also what numbers are printed on the printer and in what order?

What number(s) is(are) printed on the printer. (Also do A, B, C)

```
       8   12
       DATA DIVISION.
       WORKING-STORAGE SECTION.
       01  THE-CARD-IN    PICTURE 9(5)
       PROCEDURE DIVISION.
       START.
           ACCEPT THE-CARD-IN.
       LOOP-OUTPT.
           IF THE-CARD-IN IS GREATER THAN 25 SUBTRACT 9
           FROM THE-CARD-IN GIVING THE-CARD-IN
           DIVIDE 10 INTO THE-CARD-IN
           DISPLAY THE-CARD-IN
           GO TO FINISH.
           ADD 5 TO THE-CARD-IN.
           GO TO LOOP-OUTPT.
       FINISH.
           STOP RUN.
```

```
ccl
 00004 ← THE-CARD-IN
```

APPENDIX I

Samples of Student Work - Data Processing Course

```
                         H-200 COBOL F COMPILATION RECORD

 EXTERNAL IDENTIFICATION= AVERYE    RUN NO:        RUN DATE: 07/29/70

 OBJECT PROGRAM VISIBILITY:  VISIBLE TO ALL.

COMPILER VERSION: 01.0        VISIBILITY: L       SOURCE PROG. AND LIBR. TAPE NAME: UNUSED      MSPLT:

THERE ARE       1 WARNING DIAGNOSTICS.

OBJECT TIME MEMORY REQUIREMENT IS 012537 (005471 DECIMAL) CHARACTERS.
```

PROJECT SOUL

Completed 7/28/70

INSTRUCTOR: NORM GALE

```
CCMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= AVERYE    RUN NO:        RUN DATE: 07/29/70         PAGE: 001

DIAGS     SEQ NO    LIB NO  C                        COBOL SOURCE LANGUAGE STATEMENT                              OBJ LOC

          000010    0000       IDENTIFICATION DIVISION.                                               TEAM4
          000020    0000       PROGRAM-ID. TWO.                                                       TEAM4.
          000030    0000       AUTHOR. ELIA.                                                          TEAM4
          000040    0000       ENVIRONMENT DIVISION.                                                  TEAM4
          000050    0000       CONFIGURATION SECTION.                                                 TEAM4
          000060    0000       SOURCE-COMPUTER. H-200-SPECIAL.                                        TEAM4
          000070    0000       OBJECT-COMPUTER. H-200, SUPERVISOR,                                    TEAM4
          000080    0000           NO SEGMENTATION.                                                   TEAM4
          000090    0000       DATA DIVISION.                                                         TEAM4
          000100    0000       WORKING-STORAGE SECTION.                                               TEAM4
          000110    0000       77  TOT-COST             PICTURE 9999V99.                              TEAM4       004124
          000120    0000       01  INPUT-DATA-CARD.                                                   TEAM4       004133
          000130    0000           02  POUNDS           PICTURE 999.                                  TEAM4       004127
          000140    0000           02  COST-PER-POUND    PICTURE 99V99.                               TEAM4       004133
          000150    0000       01  OUT-PUT-DC.                                                        TEAM4       004270
          000160    0000           02  FILLER           PICTURE X(31) VALUE SPACES.                   TEAM4       004172
          000170    0000           02  PPOUNDS          PICTURE ZZ9.                                  TEAM4       004175
          000180    0000           02  FILLER           PICTURE X(23) VALUE SPACES.                   TEAM4       004224
          000190    0000           02  UNIT-COST        PICTURE $$9.99.                               TEAM4       004232
          000200    0000           02  FILLER           PICTURE X(22) VALUE SPACES.                   TEAM4       004260
          000210    0000           02  TOTAL-COST       PICTURE $$$$9.99.                             TEAM4       004270
          000220    0000       01  HEADING-I.                                                         TEAM4       004427
          000230    0000           02  FILLER           PICTURE X(30) VALUE SPACES.                   TEAM4.      004326
          000240    0000           02  HEAD-COL-I       PICTURE X(6) VALUE :POUNDS:.                  TEAM4       004334
          000250    0000           02  FILLER           PICTURE X(20) VALUE SPACES.                   TEAM4       004360
          000260    0000           02  HEAD-COL-2       PICTURE X(9) VALUE :UNIT COST:.               TEAM4       004371
          000270    0000           02  FILLER           PICTURE X(20) VALUE SPACES.                   TEAM4       004415
          000280    0000           02  HEAD-COL-3       PICTURE X(10) VALUE :TOTAL COST:.             TEAM4       004427
          000290    0000       PROCEDURE DIVISION.                                                    TEAM4
          000299    0505       /BEGIN/PROG SECTION 0.
          000300    0000       PAR-1.                                                                 TEAM4
          000310    0000           DISPLAY HEADING-I.                                                 TEAM4
          000320    0000       PAR-2.                                                                 TEAM4
          000330    0000           ACCEPT INPUT-DATA-CARD.                                            TEAM4
          000340    0000           IF POUNDS IS EQUAL TO 0 STOP :JOB:.                                TEAM4
          000350    0000           MULTIPLY POUNDS BY COST-PER-POUND GIVING TOT-COST.                 TEAM4
                                                                            *1
 1  WARNING:  HIGH ORDER DIGITS WILL BE TRUNCATED AT OBJECT TIME.                                    3  7  000033

          000360    0000           MOVE POUNDS TO PPOUNDS.                                            TEAM4
          000370    0000           MOVE COST-PER-POUND TO UNIT-COST.                                  TEAM4
          000380    0000           MOVE TOT-COST TO TOTAL-COST.                                       TEAM4
          000390    0000           DISPLAY OUT-PUT-DC.                                                TEAM4
          000400    0000           GO TO PAR-2.                                                       TEAM4
          999098    0505       BEGIN/PROG/AT /BEGIN/PROG
          999099    0001       END COBOL
```

```
   COMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= AVERYE    RUN NO:        RUN DATE: 07/29/70          PAGE: 002

    ADDRESSES OF SUBROUTINES INCLUDED IN OBJECT PROGRAM.

  ADDRESS    NAME      FUNCTION

   004651   MULT       MULTIPLY EXECUTION.
   005134   EXIT1      I/O WORK AREA.
   005633   ACCEPT     ACCEPT EXECUTION.
   006027   DSPPRT     DISPLAY EXECUTION.
   006221   ASIGN3     DYNAMIC CHANNEL ASSIGNMENT.
   006550   DELAY      DEVICE BUSY DELAY.
   006655   START      START AND SCRAP CONTROL.
   007052   FILE01     INITIALIZATION.
   007122   FILE11     DEVICE REASSIGNMENT.
   010373   FILE21     DEVICE REASSIGNMENT.
   011536   STOPIT     SCRAPPING HALTS.
                               POUNDS                  UNIT COST                  TOTAL COST


                                 11                      $1.25                      $13.75


                                 12                      $1.25                      $15.00


                                 13                      $1.25                      $16.25


                                 14                      $1.25                      $17.50


                                 15                      $1.25                      $18.75


                                 16                      $1.25                      $20.00


                                 17                      $1.25                      $21.25


                                 18                      $1.25                      $22.50
```

OK
7/28/70
D. E. Gale

```
CCMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= AVERYE    RUN NO:        RUN DATE: 07/28/70          PAGE: 001

DIAGS     SEQ NO    LIB NO  C                          COBOL SOURCE LANGUAGE STATEMENT                          OBJ LOC

          000010     0000        IDENTIFICATION DIVISION.                                       TEAM2
          000020     0000        PROGRAM-ID. NELLA-ILMO-JAY.                                    TEAM2
          000030     0000        AUTHOR. JENKINS-MASON-TIMS.                                    TEAM2
          000040     0000        ENVIRONMENT DIVISION.                                          TEAM2
          000050     0000        CONFIGURATION SECTION.                                         TEAM2
          000060     0000        SOURCE-COMPUTER. H-200-SPECIAL.                                TEAM2
          000070     0000        OBJECT-COMPUTER. H-200, SUPERVISOR, NO SEGMENTATION.           TEAM2
          000080     0000        INPUT-OUTPUT SECTION.                                          TEAM2
          000090     0000        FILE-CONTROL.                                                  TEAM2
          000100     0000            SELFCT IN-FILE ASSIGN TO CARD-READEF E.                    TEAM2
                                               *1
  1  OBSERVE:  THIS FILE-NAME IS REPRESENTED IN OBJECT TIME PRINTOUTS AS "FILE 1".                 1  3  000101

          000110     0000            SELECT OUT-FILE ASSIGN TO PRINTER R.                       TEAM2
                                               *1
  1  OBSERVE:  THIS FILE-NAME IS REPRESENTED IN OBJECT TIME PRINTOUTS AS "FILE 2".                 1  3  000102

          000120     0000        I-O-CONTROL.                                                   TEAM2
          000130     0000            APPLY H-200-SPECIAL ON IN-FILE.                            TEAM2
          000140     0000        DATA DIVISION.                                                 TEAM2
          000150     0000        FILE SECTION.                                                  TEAM2
          000160     0000        FD IN-FILE                                                     TEAM2
          000170     0000            LABEL RECORDS ARE OMITTED.                                 TEAM2
          000180     0000            VALUE OF IDENTIFICATION IS : :.                            TEAM2
          000190     0000            DATA RECORD IS IN-DATA-RECORD.                             TEAM2
          000200     0000        01  IN-DATA-RECORD.                                            TEAM2        004740
          000210     0000            02  FILLER                    PICTURF X(20).               TEAM2        004644
          000220     0000            02  NELLA-NO-IN               PICTURF 9(3).                TEAM2        004647
          000230     0000            02  FILLER                    PICTURF X(7).                TEAM2        004656
          000240     0000            02 ILMO-NO-IN                 PICTURF 9(4).                TEAM2        004662
          000250     0000            02  FILLER                    PICTURF X(16).               TEAM2        004702
          000260     0000            02  JAY-NO-IN                 PICTURE 9(4).                TEAM2        004706
          000270     0000            02  FILLER                    PICTURF X(26).               TEAM2        004740
          000280     0000        FD  OUT-FILE                                                   TEAM2
          000290     0000            LABEL RECORDS ARE OMITTED                                  TEAM2
          000300     0000            DATA RECORD IS OUT-DATA-RECORD.                            TEAM2
          000310     0000        01  OUT-DATA-RECORD.                                           TEAM2        005323
          000320     0000            02  FILLER                    PICTURF X(28).               TEAM2        005167
          000330     0000            02  NELLA-NO-OUT              PICTURF 9(3).                TEAM2        005172
          000340     0000            02  FILLER                    PICTURF X(33).               TEAM2        005233
          000350     0000            02  ILMO-NO-OUT               PICTURF 9(4).                TEAM2        005237
          000360     0000            02  FILLER                    PICTURF X(26).               TEAM2        005271
          000370     0000            02  JAY-NO-OUT                PICTURF 9(4).                TEAM2        005275
          000380     0000            02  FILLER                    PICTURF X(22).               TEAM2        005323
          000390     0000        WORKING-STORAGE SECTION.                                       TEAM2
          000400     0000        01  HEADINGS.                                                  TEAM2        004306
          000410     0000            02  FILLER                    PICTURE X(21) VALUE SPACES.  TEAM2        004143
          000420     0000            02  HEADING-1   PICTURE X(18) VALUE :JENKINS-MASON-TIMS:.  TEAM2        004165
          000430     0000            02  FILLER                    PICTURE X(21) VALUE SPACES.  TEAM2        004212
          000440     0000            02  HEADING-2   PICTURE X(12) VALUE :CASE STUDY 5:.        TEAM2        004226
          000450     0000            02  FILLER                    PICTURE X(21) VALUE SPACES.  TEAM2        004253
          000460     0000            02  HEADING-3   PICTURE X(5) VALUE :COBOL:.                TEAM2        004260
          000470     0000            02  FILLER                    PICTURF X(22) VALUE SPACES.  TEAM2        004306
          000480     0000        01  COL-HEADINGS.                                              TEAM2        004476
```

OK
Recorded 7/31/70
Norm Gale

```
COMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= AVERYE    RUN NO:        RUN DATE: 07/28/70           PAGE: 002    06

DIAGS     SEQ NO    LIB NO  C                          COBOL SOURCE LANGUAGE STATEMENT                                OBJ LOC

          000490    0000            02  FILLER                  PICTURE X(27) VALUE SPACES.        TEAM2          004341
          000500    0000            02  COL-HEADING-1           PICTURE X(5) VALUE :NELLA:.        TEAM2          004346
          000510    0000            02  FILLER                  PICTURE X(32) VALUE SPACES.        TEAM2          004406
          000520    0000            02  COL-HEADING-2           PICTURE X(4) VALUE :ILMO:.         TEAM2          004412
          000530    0000            02  FILLER                  PICTURE X(26) VALUE SPACES.        TEAM2          004444
          000540    0000            02  COL-HEADING-3           PICTURE X(3) VALUE :JAY:.          TEAM2          004447
          000550    0000            02  FILLER                  PICTURE X(23) VALUE SPACES.        TEAM2          004476
          000560    0000        PROCEDURE DIVISION.                                                TEAM2
          000569    0505        /BEGIN/PROG SECTION 0.
          000570    0000        START. OPEN INPUT IN-FILE, OUTPUT OUT-FILE.                        TEAM2
          000580    0000        HEADINGS.                                                          TEAM2
          000590    0000            DISPLAY HEADINGS.                                              TEAM2
          000600    0000        COL-HEADINGS.                                                      TEAM2
          000610    0000            DISPLAY COL-HEADINGS.                                          TEAM2
          000620    0000        RE-AD.                                                             TEAM2
          000630    0000            READ IN-FILE AT END GO TO EOJ.                                 TEAM2
          000640    0000            MOVE SPACES TO OUT-DATA-RECORD.                                TEAM2
          000650    0000            MOVE NELLA-NO-IN TO NELLA-NO-OUT.                              TEAM2
          000660    0000            MOVE ILMO-NO-IN TO ILMO-NO-OUT.                                TEAM2
          000670    0000            MOVE JAY-NO-IN TO JAY-NO-OUT.                                  TEAM2
          000680    0000            WRITE OUT-DATA-RECORD.                                         TEAM2
          000690    0000            GO TO RE-AD.                                                   TEAM2
          000700    0000        EOJ.                                                               TEAM2
          000710    0000            CLOSE IN-FILE, OUT-FILE.                                       TEAM2
          000720    0000            STOP :JOB:.                                                    TEAM2
          999998    0505        BEGIN/PROG/AT /BEGIN/PROG
          999999    0001        END COBOL
```

```
COMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= AVERYE    RUN NO:         RUN DATE: 07/28/7

   ADDRESSES OF SUBROUTINES INCLUDED IN OBJECT PROGRAM.

 ADDRESS    NAME     FUNCTION

  005535   EXIT1     I/O WORK AREA.
  006234   OPCLRT    OPEN-CLOSE CONTROL.
  006456   DSPPRT    DISPLAY EXECUTION.
  006650   ASIGN3    DYNAMIC CHANNEL ASSIGNMENT.
  007177   DELAY     DEVICE BUSY DELAY.
  007304   TAPERR    TAPE ERROR HANDLING.
  010317   PERERR    PERIPHERAL ERROR HANDLING.
  010771   TAPEOT    END-OF-TAPE CONTROL.
  011137   START     START AND SCRAP CONTROL.
  011334   OPENFR    GENERAL OPEN,ALL FILES.
  012073   TPOPEN    GENERAL OPEN, TAPE FILES.
  012751   TBEGLB    TAPE FILE BEGIN LABEL WRITE.
  014064   TLBCHK    TAPE FILE LABEL CHECK.
  014727   PRINTO    GENERAL OPEN, PRINT FILES.
  015365   CARDOP    GENERAL OPEN, CARD READ FILES.
  016053   CLOSFR    GENERAL CLOSE, ALL FILES.
  016676   TPCLS1    TAPE FILE BUFFER CLOSE.
  017342   TENDL1    TAPE FILE END LABEL WRITE.
  020150   TPCLS2    TAPE REEL SWAP.
  020757   FILE01    INITIALIZATION.
  021027   FILE11    DEVICE REASSIGNMENT.
  022300   FILE21    DEVICE REASSIGNMENT.
  023443   STOP11    SCRAPPING HALTS.
```

```
COMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= AVERYE    RUN NO:        RUN DATE: 07/28/70         PAGE: 004

  GENERAL INFORMATION ON OBJECT FILES.

FILE  TABLE  SUBR.       BUFFER  ADDRESSES          RECORD ADDR.  TYPE         DEVICES   L  IDENT. DATE    M-F-T    S.A.P.
NO.   ADDR.  ADDR.  LEFT  RIGHT  LEFT     RIGHT  LEFT   RIGHT                 1ST  2ND     ADDR.  ADDR. CONTROLS  SUBR.

 1    024214 024350 004500 004617                 004621 004740  A CARD IN   41         0

 2    024440 024574 004743 005132                 005134 005323  A PRINT     02         0

                   JENKINS-MASON-TIMS                              CASE STUDY 5                      COBOL


                        NELLA                                       ILMO                             JAY



                         203                                        3042                             5794
                         053                                        3720                             2468
                         570                                        9436                             7636
                         802                                        8026                             6401
                         779                                        0328                             9820
                         012                                        3456                             5678
                         112                                        5533                             4400
                         889                                        4452                             2011
                         852                                        4561                             7502
                         234                                        8542                             3568
                         756                                        8236                             5069
```

OK
7/28/70
N.E. Gale

```
                              H-200 COBOL F COMPILATION RECORD

 EXTERNAL IDENTIFICATION= BARAJA    RUN NO:        RUN DATE: 07/30/70

 OBJECT PROGRAM VISIBILITY:  VISIBLE TO ALL.

COMPILER VERSION: 01.0        VISIBILITY: L        SOURCE PROG. AND LIBR. TAPE NAME: UNUSED        MSPLT:

OBJECT TIME MEMORY REQUIREMENT IS 026640 (011680 DECIMAL) CHARACTERS.
```

PROJECT SOUL

EXTRA CREDIT

COBOL CASE STUDY

```
COMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= BARAJA    RUN NO:       RUN DATE: 07/30/70          PAGE: 001

DIAGS     SEQ NO   LIB NO  C                        COBOL SOURCE LANGUAGE STATEMENT                                OBJ LOC

          000010    0000          IDENTIFICATION DIVISION.                                        TEAM7
          000020    0000          PROGRAM-ID. [illegible]-CREDIT.                                 TEAM7
          000030    0000          AUTHOR. TEAM7-VICTOR-BARAJAS.                                   TEAM7
          000040    0000          ENVIRONMENT DIVISION.                                           TEAM7
          000050    0000          CONFIGURATION SECTION.                                          TEAM7
          000060    0000          SOURCE-COMPUTER. H-200-SPECIAL.                                 TEAM7
          000070    0000          OBJECT-COMPUTER. H-200. SUPERVISOR. NO SEGMENTATION.            TEAM7
          000080    0000          SPECIAL-NAMES.                                                  TEAM7
          000090    0000              PAGE IS NEXT-PAGE.                                          TEAM7
          000100    0000          INPUT-OUTPUT SECTION.                                           TEAM7
          000110    0000          FILE-CONTROL.                                                   TEAM7
          000120    0000              SELECT IN-FILE ASSIGN TO CARD-READER E.                     TEAM7
                                                *1
1  OBSERVE:  THIS FILE-NAME IS REPRESENTED IN OBJECT TIME PRINTOUTS AS "FILE 1".                    1  3  000101

          000130    0000              SELECT OUT-FILE ASSIGN TO PRINTER B.                        TEAM7
                                                *1
1  OBSERVE:  THIS FILE-NAME IS REPRESENTED IN OBJECT TIME PRINTOUTS AS "FILE 2".                    1  3  000102

          000140    0000          I-O-CONTROL.                                                    TEAM7
          000150    0000              APPLY H-200-SPECIAL ON IN-FILE.                             TEAM7
          000160    0000          DATA DIVISION.                                                  TEAM7
          000170    0000          FILE SECTION.                                                   TEAM7
          000180    0000          FD  IN-FILE                                                     TEAM7
          000190    0000          LABEL RECORDS ARE STANDARD                                      TEAM7
          000200    0000          VALUE OF IDENTIFICATION IS : :                                  TEAM7
          000210    0000          DATA RECORDS IS EMP-REC.                                        TEAM7
          000220    0000          01  EMP-REC.                                                    TEAM7          004722
          000230    0000              02 IDNO                     PICTURE 99999.                  TEAM7          004607
          000240    0000              02 INNAME.                                                  TEAM7          004640
          000250    0000                  03 INLAST               PICTURE A(15).                  TEAM7          004626
          000260    0000                  03 INFIRST              PICTURE A(9).                   TEAM7          004637
          000270    0000                  03 INMI                 PICTURE A(1).                   TEAM7          004640
          000280    0000              02  INSSNO.                                                 TEAM7          004651
          000290    0000                  03 ONE                  PICTURE 999.                    TEAM7          004643
          000300    0000                  03 TWO                  PICTURE 99.                     TEAM7          004645
          000310    0000                  03 THREE                PICTURE 9999.                   TEAM7          004651
          000320    0000              02  INOCODE                 PICTURE 99.                     TEAM7          004653
          000330    0000              02  FILLER                  PICTURE X(8).                   TEAM7          004663
          000340    0000              02  INPARATE                PICTURE 99V999.                 TEAM7          004670
          000350    0000              02  FILLER                  PICTURE X(4).                   TEAM7          004674
          000360    0000              02  INDEPT                  PICTURE 99.                     TEAM7          004676
          000370    0000              02  FILLER                  PICTURE X(20).                  TEAM7          004722
          000380    0000          FD  OUT-FILE                                                    TEAM7
          000390    0000          LABEL RECORDS ARE STANDARD                                      TEAM7
          000400    0000          VALUE OF IDENTIFICATION IS : :                                  TEAM7
          000410    0000              DATA RECORDS IS OUT-REC.                                    TEAM7
          000420    0000          01  OUT-REC                     PICTURE X(120).                 TEAM7          005305
          000430    0000          WORKING-STORAGE SECTION.                                        TEAM7
          000440    0000          77  COUNTER                     PICTURE 99.                     TEAM7          004120
          000450    0000          01  PRINT-LINE-1.                                               TEAM7          004270
          000460    0000              02  FILLER                  PICTURE XXXXXX VALUE SPACES.    TEAM7          004126
          000470    0000              02 ID-OUT                   PICTURE 99999.                  TEAM7          004133
          000480    0000              02  FILLER                  PICTURE XXXXX VALUE SPACES.     TEAM7          004140
```

OK
Recorded
7/30/70
Norm Dale

125

126

```
COMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= BARAJA     RUN NO:        RUN DATE: 07/30/70          PAGE: 002    33

DIAGS     SEQ NO    LIB NO  C                     COBOL SOURCE LANGUAGE STATEMENT                                  OBJ LOC

          000490    0000          02 SSOUT.                                                    TEAM7         004153
          000500    0000              03 SSONE                   PICTURE 999.                  TEAM7         004143
          000510    0000              03 HYP-1                   PICTURE X.                    TEAM7         004144
          000520    0000              03 SSTWO                   PICTURE 99.                   TEAM7         004146
          000530    0000              03 HYP-2                   PICTURE X.                    TEAM7         004147
          000540    0000              03 SSTHREE                 PICTURE 9999.                 TEAM7         004153
          000550    0000          02  FILLER                     PICTURE XXXXX VALUE SPACES.   TEAM7         004160
          000560    0000      02  OUT-NAME.                                                    TEAM7         004270
          000570    0000              03  OUTFIRST               PICTURE A(9).                 TEAM7         004171
          000580    0000              03  FILLER                 PICTURE X VALUE SPACES.       TEAM7         004172
          000590    0000              03  OUTMI                  PICTURE A.                    TEAM7         004173
          000600    0000          03  PERIOD                     PICTURE X.                    TEAM7         004174
          000610    0000              03  FILLER                 PICTURE X VALUE SPACES.       TEAM7         004175
          000620    0000          03  OUTLAST                        PICTURE X(59).            TEAM7         004270
          000630    0000      01  PRINT-LINE-2.                                                TEAM7         004460
          000640    0000          02  FILLER                      PICTURE XXXXXX VALUE SPACES. TEAM7         004276
          000650    0000          02  OUTOCODE                   PICTURE 99.                   TEAM7         004300
          000660    0000          02  FILLER                     PICTURE X(8) VALUE SPACES.    TEAM7         004310
          000670    0000          02  OUTDEPT                    PICTURE 99.                   TEAM7         004312
          000680    0000          02  FILLER                     PICTURE X(14) VALUE SPACES.   TEAM7         004330
          000690    0000          02  OUTPARATE                  PICTURE ZZ.999.               TEAM7         004336
          000700    0000          02  FILLER                     PICTURE X(82) VALUE SPACES.   TEAM7         004460
          000710    0000      PROCEDURE DIVISION.                                              TEAM7
          000719    0505      /BEGIN/PROG SECTION 0.
          000720    0000      HSKPG.  OPEN INPUT IN-FILE, OUTPUT OUT-FILE.                     TEAM7
          000730    0000      PAR-2.  READ IN-FILE AT END GO TO WRAP-UP.                       TEAM7
          000740    0000          MOVE SPACES TO OUT-REC.                                      TEAM7
          000750    0000          MOVE IDNO TO ID-OUT.                                         TEAM7
          000760    0000          MOVE ONE TO SSONE.                                           TEAM7
          000770    0000          MOVE TWO TO SSTWO.                                           TEAM7
          000780    0000          MOVE THREE TO SSTHREE.                                       TEAM7
          000790    0000          MOVE :-: TO HYP-1, HYP-2.                                    TEAM7
          000800    0000          MOVE INFIRST TO OUTFIRST.                                    TEAM7
          000810    0000          MOVE INMI TO OUTMI.                                          TEAM7
          000820    0000          MOVE :.: TO PERIOD.                                          TEAM7
          000830    0000          MOVE INLAST TO OUTLAST.                                      TEAM7
          000840    0000          MOVE PRINT-LINE-1 TO OUT-REC.                                TEAM7
          000850    0000          ADD 1 TO COUNTER.                                            TEAM7
          000860    0000          IF COUNTER IS GREATER THAN 8 WRITE OUT-REC AFTER ADVANCING   TEAM7
          000870    0000          NEXT-PAGE, MOVE ZEROES TO COUNTER                            TEAM7
          000880    0000          ELSE WRITE OUT-REC AFTER ADVANCING 3.                        TEAM7
          000890    0000          MOVE INOCODE TO OUTOCODE.                                    TEAM7
          000900    0000          MOVE INDEPT TO OUTDEPT.                                      TEAM7
          000910    0000          MOVE INPARATE TO OUTPARATE.                                  TEAM7
          000920    0000          MOVE PRINT-LINE-2 TO OUT-REC.                                TEAM7
          000930    0000          WRITE OUT-REC AFTER ADVANCING 1.                             TEAM7
          000940    0000          GO TO PAR-2.                                                 TEAM7
          000950    0000      WRAP-UP.  CLOSE IN-FILE, OUT-FILE.                               TEAM7
          000960    0000          STOP :JOB:.                                                  TEAM7
          999998    0505      BEGIN/PROG/AT /BEGIN/PROG
          999999    0001      END COBOL
```

H.O.F. Routine

N.C.Gale
OK
7/31/70

Grade A.

127

128

```
COMPILER= COBOL F 01.0      EXTERNAL IDENTIFICATION= BARAJA    RUN NO:        RUN DATE: 07/30/70

  ADDRESSES OF SUBROUTINES INCLUDED IN OBJECT PROGRAM.

ADDRESS   NAME    FUNCTION

 005532  EXIT1    I/O WORK AREA.
 006231  OPCLRT   OPEN-CLOSE CONTROL.
 006453  DSPPRT   DISPLAY EXECUTION.
 006645  ASIGN3   DYNAMIC CHANNEL ASSIGNMENT.
 007174  DELAY    DEVICE BUSY DELAY.
 007301  CLBTST   CARD FILE END LABEL TESTING.
 007450  LABTST   TAPE FILE END LABEL TESTING.
 010166  TAPERR   TAPE ERROR HANDLING.
 011201  PERERR   PERIPHERAL ERROR HANDLING.
 011653  TAPEOT   END-OF-TAPE CONTROL.
 012021  START    START AND SCRAP CONTROL.
 012216  OPENER   GENERAL OPEN,ALL FILES.
 012755  TPOPEN   GENERAL OPEN, TAPE FILES.
 013633  TBEGLB   TAPE FILE BEGIN LABEL WRITE.
 014746  TLBCHK   TAPE FILE LABEL CHECK.
 015611  PRINTO   GENERAL OPEN, PRINT FILES.
 016247  CARDOP   GENERAL OPEN, CARD READ FILES.
 016735  CLBCHK   CARD FILE LABEL CHECK.
 017326  CLOSER   GENERAL CLOSE, ALL FILES.
 020151  TPCLS1   TAPE FILE BUFFER CLOSE.
 020615  TENDLL   TAPE FILE END LABEL WRITE.
 021423  TPCLS2   TAPE REEL SWAP.
 022232  FILE01   INITIALIZATION.
 022302  FILE11   DEVICE REASSIGNMENT.
 023553  FILE21   DEVICE REASSIGNMENT.
 024716  STOPIT   SCRAPPING HALTS.
```

```
COMPILER= COBOL F 01.0     EXTERNAL IDENTIFICATION= BARAJA    RUN NO:          RUN DATE: 07/30/70           PAGE: 004

  GENERAL INFORMATION ON OBJECT FILES.

FILE  TABLE  SUBR.        BUFFER  ADDRESSES          RECORD ADDR.  TYPE          DEVICES   L  IDENT. DATE    M-F-T     S.A.P.
NO.   ADDR.  ADDR.  LEFT  RIGHT   LEFT      RIGHT   LEFT   RIGHT                1ST  2ND     ADDR.  ADDR.  CONTROLS  SUBR.

 1   025467 025623 004462 004601                    004603 004722  A CARD IN   41          S 005523 000222

 2   025717 026053 004725 005114                    005116 005305  A PRINT     02          S 005523 000222



        29384      958-72-5556     FAYE       A. HEISLER
        11         06               6.575

        29384      958-72-5556     FAYE       A. HEISLER
        11         06               6.575

        29384      958-72-5556     FAYE       A. HEISLER
        11         06               6.575

        00099      215-29-7712     VIRGINIA   M. HAYTER
        13         06               6.575

        00099      215-29-7712     VIRGINIA   M. HAYTER
        13         06               6.575

        00099      215-29-7712     VIRGINIA   M. HAYTER
        13         06               6.575

        00100      534-85-6324     ROBERT     A. HICKMAN
        09         06               6.575

        00100      534-85-6324     ROBERT     A. HICKMAN
        09         06               6.575
```

```
00100      534-85-6324     ROBERT     A. HICKMAN
09         06                6.575

60000      199-52-4287     IRVIN      V. HILLYER
06         06                6.575

60000      199-52-4287     IRVIN      V. HILLYER
06         06                6.575

60000      199-52-4287     IRVIN      V. HILLYER
06         06                6.575

60000      199-52-4287     IRVIN      V. HILLYER
06         06                6.575

70000      251-51-9801     FRED       E. HERNDON
07         06                6.575

70000      251-51-9801     FRED       E. HERNDON
07         06                6.575

70000      251-51-9801     FRED       E. HERNDON
07         06                6.575

80000      108-87-3747     MARIETTA   J. HITE
06         06                6.575
```

```
        80000      108-87-3747      MARIETTA   J. HITE
        06         06                 6.575

        80000      108-87-3747      MARIETTA   J. HITE
        06         06                 6.575

        10007      522-71-1603      RUBY       I. HEBERER
        10         06                 6.575

        10007      522-71-1603      RUBY       I. HEBERER
        10         06                 6.575

        10007      522-71-1603      RUBY       I. HEBERER
        10         06                 6.575

        10000      111-11-1111      MARY       G. GRAVES
        10         01                 2.500

        10000      111-11-1111      MARY       G. GRAVES
        10         01                 2.500

        10000      111-11-1111      MARY       G. GRAVES
END PRTOB FILE 501                    2.500
```